ACTIGENCE 교육 자료

AI 바이브코딩 입문 실습 노트

v1.0.0 · 2026-11-07 14:17 KST 기준 · 최신본 actigence.ai/library/downloads

코딩을 몰라도 AI로 결과물을 만드는 4주 과정의 회차별 실습 프롬프트, 개념, 참고 링크.

1회차

AI 도구 세팅과 첫 결과물 만들기

도구를 준비하고, AI에게 말로 시켜서 첫 결과물을 내 손으로 만든다.

준비물

- 인터넷에 연결된 노트북
- Claude를 쓸 수 있는 계정
- 결과물을 모아 둘 빈 폴더 하나
- 만들고 싶은 것에 대한 짧은 메모 (예: 내 소개 페이지, 우리 가게 안내 페이지)

AI 도구와 작업 환경 준비

AI에게 일을 맡기려면 무엇을 먼저 알려 주어야 할까요?

AI는 내가 누구이고 무엇을 하려는지 모른 채 시작합니다. 첫 대화에서 직무, 목표, 수준을 알려 주면 이후 답이 훨씬 맞아집니다.

코딩 도구의 설치 방법은 시기에 따라 바뀝니다. 공식 문서의 설치 안내를 따라 설치하고, 막히는 곳은 화면 그대로 AI에게 물어봅니다.

확인

- ☐ Claude에 로그인해 대화를 시작했다
- ☐ 결과물을 모을 폴더를 만들었다
- ☐ 공식 문서를 열어 설치 안내를 확인했다

P1 · 나를 소개하고 목표 전하기

첫 대화에서 AI가 내 상황을 이해하게 만듭니다.

> 나는 [직무]를 맡고 있고, 코딩은 해 본 적이 [없어 / 조금 있어]. 이번 4주 동안 AI로 [만들고 싶은 것]을 만들어 보려고 해. 내 수준에 맞춰 쉬운 말로 설명해 줘. 모르는 용어가 나오면 한 줄로 풀어서 알려 줘.

P2 · 작업 폴더 정하기

결과물을 둘 폴더 이름과 구조를 정합니다.

> AI로 만든 결과물을 한곳에 모으려고 해. 폴더 이름은 영어 소문자와 하이픈만 써서 정하고 싶어. [만들고 싶은 것]에 어울리는 폴더 이름 후보를 3개 알려 주고, 폴더를 어디에 만들면 나중에 찾기 쉬운지도 설명해 줘.

P3 · 막힌 화면 물어보기

설치나 실행 중 나온 메시지를 AI에게 그대로 보여 주고 해결 방법을 묻습니다.

> [도구 이름]을 설치하다가 아래 메시지가 나왔어. 무슨 뜻인지 쉬운 말로 설명하고, 내가 확인해야 할 것을 순서대로 알려 줘. 확신이 없는 부분은 공식 문서에서 확인하라고 말해 줘.
>
> [화면에 나온 메시지를 그대로 붙여넣기]

> AI에게 먼저 상황을 알려 주면 이후의 모든 답이 달라집니다.

첫 결과물 만들기

한 문장으로 시킨 일이 화면으로 나오려면 무엇을 적어야 할까요?

처음에는 완성도보다 결과가 눈에 보이는 것이 중요합니다. 한 장짜리 웹 페이지처럼 브라우저에서 바로 열리는 결과물이 좋습니다.

무엇을, 누구에게, 어떤 느낌으로 보여 줄지를 적으면 AI가 빈칸을 덜 추측합니다.

P4 · 만들 것 정하기

첫 결과물의 범위를 작게 좁힙니다.

> 나는 [직무]이고 [만들고 싶은 것]을 만들어 보고 싶어. 코딩을 모르는 사람이 오늘 안에 끝낼 수 있는 크기로 줄여 줘. 선택지를 3개 주고, 각각 만드는 데 걸리는 대략의 시간과 결과 모습을 한 줄씩 적어 줘.

P5 · 한 장짜리 페이지 만들기

브라우저에서 열어 볼 수 있는 첫 페이지를 만듭니다.

> [주제]를 소개하는 한 장짜리 웹 페이지를 index.html 파일 하나로 만들어 줘. 제목, 두세 문장의 소개, 연락 방법을 넣어 줘. 글자는 읽기 편한 크기로 하고, 색은 [차분한 / 밝은] 느낌으로 맞춰 줘. 만든 뒤에 파일을 어떻게 열어 보는지도 알려 줘.

내용만 바꿔 다시 요청

> 같은 구조는 두고 소개 문장과 연락 방법만 아래 내용으로 바꿔 줘.
>
> [새로 넣을 내용]

결과: index.html

P6 · 만들어진 파일 설명 듣기

AI가 만든 결과물이 무엇으로 이뤄졌는지 이해합니다.

> 방금 만든 파일의 구조를 코딩을 모르는 사람에게 설명하듯 알려 줘. 제목, 본문, 색을 바꾸고 싶을 때 파일의 어느 부분을 고치면 되는지 짚어 줘.

산출물 index.html

> 작게 시작한 결과물이 이후 모든 수정의 출발점이 됩니다.

고치고 되돌리기

마음에 들지 않는 부분을 어떻게 말해야 정확히 고쳐질까요?

한 번에 하나씩 고치도록 요청하면 무엇이 바뀌었는지 확인하기 쉽습니다.

결과가 나빠졌다면 되돌리면 됩니다. 수정하기 전에 파일을 복사해 두면 안심하고 시도할 수 있습니다.

P7 · 한 가지씩 수정 요청

바뀐 부분을 확인하며 결과물을 다듬습니다.

> 방금 만든 [파일 이름]에서 [바꾸고 싶은 부분] 하나만 바꿔 줘. 다른 부분은 그대로 두고, 무엇을 어떻게 바꿨는지 한두 줄로 알려 줘.

P8 · 되돌리기 준비

수정 전 상태를 남겨 두는 방법을 배웁니다.

> [파일 이름]을 고치기 전에 지금 상태를 남겨 두고 싶어. 코딩을 모르는 사람도 할 수 있는 가장 쉬운 방법을 알려 줘. 나중에 원래대로 돌리는 방법도 같이 알려 줘.

P9 · 오늘 작업 정리

배운 것과 다음 주에 할 일을 기록으로 남깁니다.

> 오늘 우리가 한 작업을 notes.md 파일로 정리해 줘. 만든 파일 이름, 내가 쓴 지시 중 잘 먹힌 것, 막혔던 점, 다음 주에 해 보고 싶은 것을 항목별로 적어 줘.

결과: notes.md

산출물 notes.md

> 한 번에 하나씩 고치고, 수정 전 상태를 남겨 둡니다.

개념

프롬프트

AI에게 하는 지시문 AI에게 전달하는 질문이나 지시 전체를 말합니다. 무엇을 원하는지, 어떤 형식으로 받고 싶은지를 구체적으로 적을수록 원하는 결과에 가까워집니다.

컨텍스트

AI가 참고하는 배경 정보 AI가 답을 만들 때 참고하는 대화 내용과 자료를 말합니다. 내 직무와 목표를 먼저 알려 주면 같은 질문에도 더 알맞은 답이 나옵니다. 대화가 길어지면 앞부분을 놓칠 수 있어 중요한 조건은 다시 적어 줍니다.

바이브 코딩

말로 지시해 만드는 방식 코드를 직접 쓰는 대신 원하는 결과를 자연어로 설명하고, AI가 만든 코드를 실행해 보며 다듬어 가는 작업 방식입니다. 결과를 확인하고 고쳐 달라고 말하는 일은 사람이 맡습니다.

AI 에이전트

스스로 여러 단계를 수행하는 AI 질문에 답하는 데서 그치지 않고, 파일을 읽고 만들고 실행하는 등 여러 단계를 이어서 수행하는 AI를 말합니다. 하는 일의 범위가 넓은 만큼 무엇을 허용할지 사람이 확인해야 합니다.

터미널

글자로 컴퓨터에 명령하는 창 마우스 대신 글자로 컴퓨터에 명령을 내리는 프로그램입니다. 코딩용 AI 도구는 이 창 안에서 동작하기도 합니다. 어려워 보여도 AI가 알려 주는 명령을 하나씩 확인하며 쓰면 됩니다.

폴더와 파일 경로

파일이 놓인 위치를 가리키는 주소 파일이 어느 폴더 안에 있는지 나타내는 주소입니다. 작업 폴더를 정해 두면 AI와 나 모두 같은 위치에서 파일을 찾을 수 있습니다. 폴더 이름은 영어 소문자와 하이픈을 쓰면 문제가 적습니다.

반복 개선

만들고 확인하고 고치기를 되풀이 한 번에 완성하려 하지 않고 결과를 보고 고쳐 달라고 요청하는 과정을 되풀이하는 것입니다. 한 번에 한 가지만 바꾸면 무엇이 효과가 있었는지 알 수 있습니다.

참고 링크

- Claude — 대화로 질문하고 문서를 다듬는 Claude 웹 서비스.
- Claude Code 소개 — 터미널에서 코드를 읽고 쓰고 고치는 Claude Code가 무엇인지 소개합니다.
- Claude Code 문서 — 설치와 사용법을 다루는 공식 문서. 설치는 이 문서의 안내를 따릅니다.
- 프롬프트 엔지니어링 가이드 — 지시를 구체적으로 쓰는 방법을 정리한 공식 가이드.

2회차

프롬프트로 원하는 화면 만들기

화면을 말로 설명하면 버튼, 입력 폼, 목록이 만들어진다.

준비물

- 1주 차에 만든 작업 폴더
- 브라우저에서 열어 볼 수 있는 index.html
- 화면에 넣고 싶은 기능 한 가지 (예: 방문 예약 폼, 할 일 목록)

화면 구성을 말로 정리

만들기 전에 어떤 화면이 필요한지 먼저 말로 그려 보았나요?

화면을 어떻게 만들지 AI에게 바로 맡기기보다, 먼저 무엇이 보여야 하는지 목록으로 정리합니다.

AI가 질문을 먼저 하게 하면 내가 놓친 조건을 찾을 수 있습니다.

P1 · 화면 구성 인터뷰

AI의 질문에 답하며 화면에 필요한 요소를 정합니다.

> 나는 [무엇을 하는 화면]을 만들고 싶어. 만들기 전에 필요한 것을 알아야 하니 질문을 한 번에 하나씩 해 줘. 화면에 보여야 할 것, 사용자가 누를 것, 입력할 것 순서로 물어봐 줘. 대답이 모이면 화면 구성안을 목록으로 정리해 줘.

P2 · 요구사항 목록 만들기

만들 기능을 필수와 나중에 할 일로 나눕니다.

> 위에서 정리한 화면 구성안을 '꼭 필요한 것'과 '나중에 해도 되는 것'으로 나눠 줘. 꼭 필요한 것은 오늘 안에 만들 수 있는 양으로 줄여서 번호를 붙여 줘.

> 만들기 전에 필요한 요소를 목록으로 정리하면 수정 횟수가 줄어듭니다.

버튼과 입력 폼

버튼을 누르면 무슨 일이 일어나야 하는지 정확히 말할 수 있나요?

버튼은 누르는 순간 일어날 일까지 함께 말해야 합니다. '버튼을 만들어 줘'보다 '누르면 입력한 내용이 아래 목록에 추가되게 해 줘'가 정확합니다.

입력 폼은 어떤 항목이 필수인지, 잘못 입력하면 어떤 안내를 보일지도 함께 정합니다.

확인

- ☐ 버튼을 눌렀을 때 일어날 일을 한 문장으로 적었다
- ☐ 필수로 받을 항목을 정했다

P3 · 동작하는 버튼 만들기

버튼을 누르면 화면이 바뀌도록 만듭니다.

> [파일 이름]에 '[버튼 이름]' 버튼을 추가해 줘. 이 버튼을 누르면 [일어나야 할 일]이 되게 해 줘. 버튼은 손가락으로 누르기 편한 크기로 하고, 눌렀다는 것을 알 수 있게 화면에 표시해 줘.

결과: index.html

P4 · 입력 폼 만들기

이름, 연락처 같은 입력 항목을 가진 폼을 만듭니다.

> [파일 이름]에 입력 폼을 추가해 줘. 항목은 [항목1], [항목2], [항목3]이고, 이 중 [필수 항목]은 꼭 입력해야 해. 제출 버튼을 누르면 입력한 내용을 화면 아래에 보여 줘.

결과: index.html

P5 · 잘못된 입력 안내

빈칸이나 틀린 형식에 친절한 안내를 보여 줍니다.

> 방금 만든 입력 폼에서 필수 항목이 비어 있거나 [연락처 / 이메일] 형식이 맞지 않으면 어떤 항목이 왜 문제인지 해당 칸 아래에 알려 줘. 안내 문구는 짧고 차분한 말투로 써 줘.

산출물 index.html

> 버튼은 눌렀을 때 일어날 일까지 함께 말합니다.

목록으로 보여 주기

입력한 내용이 쌓이면 어떤 모양으로 보여 주면 좋을까요?

목록은 추가, 삭제, 빈 상태를 함께 생각해야 쓸 만해집니다.

한 번에 모두 요청하지 말고, 보여 주기, 추가하기, 지우기 순서로 나눠 확인합니다.

P6 · 목록 보여 주기

입력한 항목이 아래로 쌓이는 목록을 만듭니다.

> 입력 폼에서 제출한 내용이 [항목 이름]별로 아래 목록에 한 줄씩 쌓이게 해 줘. 가장 최근 것이 위에 보이게 하고, 아직 아무것도 없을 때는 '[비어 있을 때 보여 줄 문구]'라고 보여 줘.

결과: index.html

P7 · 삭제와 완료 표시

목록의 항목을 지우거나 끝낸 일로 표시합니다.

> 목록의 각 줄에 '삭제' 버튼과 '완료' 표시를 추가해 줘. 완료한 항목은 글자에 취소선을 그어서 색이 아니라 모양으로도 구분되게 해 줘.

P8 · 화면 모양 다듬기

여백과 글자 크기를 읽기 편하게 정리합니다.

> 지금 화면을 휴대폰과 노트북 모두에서 읽기 편하게 다듬어 줘. 글자 크기, 여백, 버튼 간격만 조정하고 기능은 바꾸지 마. 바꾼 내용은 목록으로 알려 줘.

P9 · 휴대폰에서 점검하기

좁은 화면에서 깨지는 곳을 찾습니다.

> 이 화면을 휴대폰처럼 좁은 폭에서 볼 때 깨지거나 불편해질 곳을 점검해 줘. 문제가 있으면 위치와 이유를 알려 주고, 내가 허락하면 그때 고쳐 줘.

> 빈 상태, 추가, 삭제를 나눠서 확인하면 목록이 단단해집니다.

개념

HTML·CSS·JavaScript

웹 화면을 이루는 세 가지 언어 HTML은 화면에 무엇이 있는지, CSS는 어떻게 보이는지, JavaScript는 누르면 어떻게 움직이는지를 맡습니다. 직접 쓰지 않아도 어느 부분이 어떤 역할인지 알면 AI에게 정확히 시킬 수 있습니다.

입력 폼

사용자가 정보를 적어 내는 화면 이름, 연락처처럼 사용자가 정보를 입력하고 제출하는 화면 요소입니다. 필수 항목과 입력 형식을 미리 정해 두면 잘못된 입력을 줄일 수 있습니다.

이벤트

누르기·입력하기 같은 사용자 동작 버튼을 누르거나 글자를 입력하는 것처럼 화면에서 일어나는 사용자의 동작입니다. '이 동작이 일어나면 이렇게 해 줘'라고 말하는 것이 화면을 움직이는 기본 지시입니다.

반응형 화면

화면 크기에 맞게 바뀌는 구성 휴대폰, 태블릿, 노트북 같은 서로 다른 화면 폭에 맞춰 배치가 달라지는 화면입니다. 가장 좁은 화면에서 먼저 확인하면 문제를 빨리 찾을 수 있습니다.

입력 검증

입력값이 올바른지 확인하는 과정 빈칸이나 형식이 틀린 값을 제출하기 전에 걸러 내는 과정입니다. 무엇이 왜 잘못됐는지 짧게 알려 주면 사용자가 바로 고칠 수 있습니다.

반복 개선

만들고 확인하고 고치기를 되풀이 결과를 눈으로 확인하고 한 가지씩 고쳐 달라고 요청하는 과정입니다. 화면 작업에서는 바꾼 곳만 확인하면 되어 속도가 빠릅니다.

참고 링크

- MDN 웹 개발 학습 — HTML, CSS, JavaScript 기초를 차근차근 익히는 학습 자료.
- 프롬프트 엔지니어링 가이드 — 지시를 구체적으로 쓰는 방법을 정리한 공식 가이드.
- Claude Code 문서 — 설치와 사용법을 다루는 공식 문서. 설치는 이 문서의 안내를 따릅니다.

3회차

반복 업무 자동화 실습

매주 반복하는 일을 골라 AI에게 맡기고, 결과는 사람이 확인한다.

준비물

- 이번 주에 반복한 업무 두세 가지의 메모
- 연습용 엑셀 또는 CSV 파일 (실제 고객 정보는 지우거나 가짜 값으로 바꾼 것)
- 답장이 필요한 메일 한두 건 (개인정보를 가린 것)

반복 업무 고르기

AI에게 맡길 일과 사람이 해야 할 일을 구분했나요?

매번 같은 순서로 하는 일, 규칙이 분명한 일이 자동화에 알맞습니다. 판단과 책임이 필요한 일은 사람이 남겨 둡니다.

AI에게 자료를 줄 때는 이름, 연락처 같은 개인정보와 회사의 기밀을 먼저 지우거나 바꿉니다.

확인

- ☐ 연습용 자료에서 개인정보를 지웠다
- ☐ 자동화할 업무 한 가지를 골랐다

P1 · 반복 업무 찾기

AI로 줄일 수 있는 일을 먼저 고릅니다.

> 나는 [직무]를 맡고 있어. 이번 주에 반복해서 한 업무를 하나씩 물어봐 줘. 다 듣고 나면 AI로 줄일 수 있는 일과 그렇지 않은 일을 표로 나눠 줘. 표에는 업무, 걸리는 시간, AI로 줄일 방법을 넣어 줘.

P2 · 자동화 순서 정하기

효과가 크고 쉬운 일부터 시작하도록 순위를 매깁니다.

> 위 표에서 자동화 순서를 정해 줘. 기준은 '줄어드는 시간'과 '시작하기 쉬운 정도'야. 맨 위 업무 하나를 골라서 오늘 수업 안에 해 볼 수 있는 단계로 쪼개 줘.

> 자동화는 일을 고르는 데서 절반이 정해집니다.

엑셀·CSV 정리

표 데이터를 AI에게 맡길 때 무엇을 먼저 확인해야 할까요?

엑셀 파일을 CSV로 저장하면 AI가 읽고 처리하기 쉬워집니다. 저장 방법은 사용하는 프로그램의 도움말을 확인합니다.

AI가 정리한 결과는 반드시 원본과 몇 줄 비교해 봅니다. 줄 수가 맞는지, 값이 바뀌지 않았는지 확인합니다.

P3 · 표 상태 살펴보기

정리하기 전에 데이터의 문제를 먼저 찾습니다.

> 첨부한 [파일 이름]을 열어서 먼저 점검만 해 줘. 열 이름, 전체 줄 수, 비어 있는 칸, 같은 값이 다르게 적힌 곳(예: 띄어쓰기나 표기 차이)을 찾아서 알려 줘. 아직 파일은 고치지 마.

P4 · 정리 작업 시키기

규칙을 말하고 정리된 새 파일을 받습니다.

> [파일 이름]을 아래 규칙으로 정리해서 새 파일로 만들어 줘. 원본은 그대로 두고 새 파일 이름은 [새 파일 이름]으로 해.
> 1. [규칙1]
> 2. [규칙2]
> 3. [규칙3]
> 끝나면 바뀐 줄 수와 바꾼 내용을 요약해 줘.

결과: 정리본.csv

P5 · 결과 검증하기

정리된 파일이 맞는지 원본과 비교합니다.

> 원본과 정리본을 비교해서 줄 수가 같은지, 내가 말한 규칙 밖에서 바뀐 값이 있는지 확인해 줘. 의심스러운 줄은 원본 값과 정리본 값을 나란히 보여 줘.

P6 · 다음에도 쓸 수 있게 만들기

같은 정리를 다음에 반복할 수 있는 형태로 남깁니다.

> 지금 한 정리 작업을 다음에도 같은 방식으로 시킬 수 있게 정리-방법.md로 남겨 줘. 입력 파일의 모양, 적용한 규칙, 결과 확인 방법을 순서대로 적어 줘.

결과: 정리-방법.md

산출물 정리본.csv

> AI가 정리한 결과는 원본과 비교해 확인한 뒤에 씁니다.

메일 초안과 요약

AI가 쓴 글에서 사람이 꼭 확인해야 할 부분은 어디일까요?

AI는 그럴듯한 문장을 만들지만 사실과 다른 내용을 섞을 수 있습니다. 날짜, 금액, 이름은 사람이 직접 확인합니다.

받는 사람과 목적, 말투를 알려 주면 초안의 품질이 크게 올라갑니다.

P7 · 메일 초안 쓰기

받는 사람과 목적에 맞는 메일 초안을 받습니다.

> 아래 상황으로 메일 초안을 써 줘. 받는 사람은 [받는 사람의 관계], 목적은 [목적]이야. 말투는 [정중하게 / 간결하게], 길이는 [몇 문단]으로 해 줘. 날짜와 금액처럼 내가 확인해야 하는 내용은 [확인 필요]로 표시해 줘.
>
> [전달할 내용]

P8 · 긴 글 요약하기

긴 문서에서 핵심과 할 일을 뽑습니다.

> 아래 글을 읽고 다섯 줄 이내로 요약해 줘. 그다음 내가 해야 할 일을 번호 목록으로 따로 뽑아 줘. 글에 없는 내용은 추측해서 쓰지 마.
>
> [요약할 글]

P9 · 사실 확인 요청

AI의 답에서 확인이 필요한 부분을 스스로 짚게 합니다.

> 방금 쓴 글에서 사실과 다를 가능성이 있는 부분이나 네가 확신하지 못한 부분을 모두 짚어 줘. 숫자, 날짜, 이름, 고유명사 위주로 목록으로 알려 줘.

> AI의 초안은 시작점이고, 사실 확인은 사람의 몫입니다.

개념

자동화

반복되는 일을 규칙대로 처리하게 하는 것 사람이 매번 하던 일을 정해진 규칙에 따라 컴퓨터나 AI가 처리하게 하는 것입니다. 규칙이 분명하고 자주 반복되는 일일수록 효과가 큽니다.

CSV

쉼표로 칸을 나눈 단순한 표 파일 표의 각 칸을 쉼표로 구분해 글자로만 저장한 파일 형식입니다. 엑셀에서 저장할 수 있고, AI와 다른 프로그램이 읽기 쉽습니다. 서식이나 수식은 저장되지 않습니다.

환각(hallucination)

AI가 사실이 아닌 내용을 그럴듯하게 말하는 현상 AI가 모르는 내용을 지어내거나 사실과 다른 내용을 자신 있는 말투로 답하는 현상입니다. 숫자, 날짜, 출처는 직접 확인하고, 모르면 모른다고 말하게 지시하면 줄일 수 있습니다.

프롬프트 틀

빈칸만 바꿔 다시 쓰는 지시문 자주 하는 지시를 [대괄호] 빈칸이 있는 문장으로 저장해 두고 상황에 따라 채워 쓰는 방식입니다. 같은 품질의 결과를 반복해서 얻는 데 도움이 됩니다.

민감 정보

AI에게 보내기 전에 지워야 하는 정보 개인의 이름, 연락처, 주민등록번호, 회사의 기밀 자료처럼 밖으로 나가면 곤란한 정보입니다. 연습할 때는 가짜 값으로 바꾸고, 실제 업무 자료는 조직의 규칙을 먼저 확인합니다.

스크립트

정해진 작업을 순서대로 실행하는 짧은 프로그램 파일 정리처럼 정해진 일을 순서대로 처리하는 짧은 프로그램입니다. AI가 만들어 주고, 같은 작업이 필요할 때마다 다시 실행할 수 있습니다.

참고 링크

- 프롬프트 엔지니어링 가이드 — 지시를 구체적으로 쓰는 방법을 정리한 공식 가이드.
- Claude Code 문서 — 설치와 사용법을 다루는 공식 문서. 설치는 이 문서의 안내를 따릅니다.
- Claude 고객 지원 센터 — 계정과 사용 중 생기는 문제를 찾아보는 공식 지원 문서.

4회차

내 결과물 공유하고 마무리

만든 결과물을 웹에 올려 링크로 공유하고, 막혔을 때 AI에게 묻는 법까지 익힌다.

준비물

- 1~3주 차에 만든 결과물이 들어 있는 폴더
- 공개해도 되는 내용인지 확인한 결과물 하나
- 웹에 올릴 때 쓸 서비스의 계정 (공식 문서에서 가입 방법 확인)

결과물 점검과 정리

남에게 보여 줘도 괜찮은 상태인지 확인했나요?

공개하기 전에 개인정보, 비밀번호, 내부 자료가 파일에 들어 있지 않은지 확인합니다. 한 번 올라간 내용은 되돌리기 어렵습니다.

설명 문서를 하나 남겨 두면 나중에 나도 다른 사람도 무엇을 만든 것인지 알 수 있습니다.

확인

- ☐ 파일에 개인정보나 비밀번호가 없다
- ☐ 링크를 받을 사람이 누구인지 정했다

P1 · 공개 전 점검

올리면 곤란한 내용이 없는지 AI와 함께 훑어봅니다.

> 이 폴더의 파일을 웹에 공개하려고 해. 이름, 연락처, 비밀번호, 내부 자료처럼 공개하면 곤란한 내용이 들어 있는지 파일별로 점검해 줘. 발견하면 어느 파일 어느 부분인지 알려 주고, 고치기 전에 나에게 먼저 물어봐 줘.

P2 · 설명 문서 만들기

무엇을 만들었는지 다른 사람이 알 수 있게 남깁니다.

> 이 폴더의 결과물을 처음 보는 사람에게 설명하는 README.md를 만들어 줘. 무엇을 하는 결과물인지, 어떤 파일이 있는지, 어떻게 열어 보는지를 짧게 적어 줘.

결과: README.md

산출물 README.md

> 올리기 전에 확인하는 습관이 가장 중요한 안전장치입니다.

웹에 올리고 링크 공유

내 파일이 어떻게 모두가 열 수 있는 주소가 될까요?

정적 웹사이트는 파일을 호스팅 서비스에 올리면 주소가 생깁니다. 서비스마다 올리는 방법이 다르므로 공식 문서의 안내를 따릅니다.

올린 뒤에는 시크릿 창이나 다른 기기에서 열어 내 컴퓨터가 아니어도 보이는지 확인합니다.

P3 · 올릴 곳 고르기

내 결과물에 맞는 호스팅 방법을 비교합니다.

> 내 결과물은 [HTML 파일 하나 / 여러 파일로 된 정적 사이트]야. 코딩을 모르는 사람이 웹에 올려 링크를 만들 수 있는 방법을 두세 가지 비교해 줘. 각각 어떤 사람에게 맞는지 적고, 구체적인 메뉴 이름이나 명령은 지어내지 말고 공식 문서를 찾아보라고 말해 줘.

P4 · 올리는 순서 안내받기

고른 방법의 흐름을 순서대로 이해합니다.

> [선택한 서비스]에 내 폴더를 올려 링크를 만들려고 해. 큰 흐름을 단계로 설명해 줘. 내 화면과 설명이 다르면 화면 그대로 알려 줄 테니 그때 다시 안내해 줘. 확실하지 않은 부분은 공식 문서에서 확인하라고 알려 줘.

P5 · 올린 뒤 점검

공유 전에 링크가 제대로 열리는지 확인합니다.

> 웹에 올린 링크를 다른 사람에게 보내기 전에 확인할 목록을 만들어 줘. 다른 기기에서 열리는지, 휴대폰에서 글자가 잘 보이는지, 링크를 눌렀을 때 오류가 없는지를 포함해 줘.

> 링크를 보내기 전에 다른 기기에서 먼저 열어 봅니다.

AI에게 묻는 법과 다음 단계

막혔을 때 어떻게 물어야 답이 빨리 나올까요?

질문에는 하려던 일, 실제로 일어난 일, 이미 해 본 것을 함께 적습니다. 오류 메시지는 요약하지 말고 그대로 붙입니다.

4주 동안 효과가 좋았던 프롬프트는 모아 두면 다음 프로젝트의 출발점이 됩니다.

P6 · 막혔을 때 질문하기

문제 상황을 AI가 이해하기 쉬운 틀로 전달합니다.

> 하려던 일: [하려던 일]
> 실제로 일어난 일: [일어난 일]
> 이미 해 본 것: [해 본 것]
> 화면 또는 오류 메시지: [그대로 붙여넣기]
> 원인 후보를 가능성이 높은 순서로 알려 주고, 내가 먼저 확인할 것을 한 가지씩 알려 줘.

P7 · 내 프롬프트 모음 만들기

4주 동안 잘 먹힌 지시를 한 파일로 모읍니다.

> 우리가 이 대화에서 쓴 지시 중 결과가 좋았던 것을 prompts.md로 모아 줘. 각 지시마다 언제 쓰는지 한 줄 설명을 달고, 상황에 따라 바뀌는 부분은 [대괄호] 빈칸으로 바꿔 줘.

결과: prompts.md

P8 · 다음 프로젝트 정하기

배운 것을 이어 갈 다음 만들 것을 고릅니다.

> 나는 [직무]이고 4주 동안 [만든 것]을 만들었어. 이 경험을 바탕으로 다음에 해 볼 만한 프로젝트를 세 가지 제안해 줘. 각각 만드는 데 필요한 것, 예상되는 어려움, 첫 번째로 할 일을 적어 줘.

산출물 prompts.md

> 잘 먹힌 프롬프트를 모아 두면 다음 결과물이 빨라집니다.

개념

저장소(Git)

파일의 변경 이력을 기록하는 공간 Git은 파일이 언제 어떻게 바뀌었는지 기록해 두는 도구이고, 저장소는 그 기록을 담은 폴더입니다. 실수했을 때 이전 상태로 돌아갈 수 있고, 여러 사람이 함께 작업하기 쉬워집니다.

커밋

변경 내용을 기록으로 남기는 저장 지점 저장소에 현재 상태를 하나의 기록으로 남기는 것입니다. 무엇을 바꿨는지 짧은 설명과 함께 남기며, 나중에 그 시점으로 돌아갈 수 있습니다.

배포

만든 결과물을 다른 사람이 쓸 수 있게 올리는 일 내 컴퓨터에 있는 결과물을 인터넷의 서버에 올려 누구나 주소로 열 수 있게 하는 일입니다. 올리기 전에 공개해도 되는 내용인지 확인합니다.

정적 웹사이트

만들어 둔 파일을 그대로 보여 주는 사이트 HTML, CSS, 이미지처럼 미리 만들어 둔 파일을 그대로 보여 주는 웹사이트입니다. 서버에서 계산하거나 저장하는 기능이 없어 올리기 쉽고, 입문 단계의 결과물에 알맞습니다.

API

프로그램끼리 정보를 주고받는 약속 한 프로그램이 다른 프로그램에 기능이나 정보를 요청하고 받는 방식을 정해 둔 약속입니다. 날씨나 지도 같은 외부 기능을 내 결과물에 붙일 때 쓰며, 다음 단계에서 다루게 됩니다.

환각(hallucination)

AI가 사실이 아닌 내용을 그럴듯하게 말하는 현상 AI가 없는 서비스 이름이나 메뉴를 지어내 알려 줄 수 있습니다. 설치와 배포처럼 정확해야 하는 안내는 공식 문서와 대조해 확인합니다.

참고 링크

- Vercel 문서 — 만든 사이트를 웹에 올리는 호스팅 서비스의 공식 문서.
- GitHub Pages — 저장소의 정적 파일을 무료로 웹에 공개하는 서비스 안내.
- Pro Git 한국어판 — 저장소와 커밋 개념을 처음부터 설명하는 무료 책.
- MDN 웹 개발 학습 — HTML, CSS, JavaScript 기초를 차근차근 익히는 학습 자료.
- Claude 고객 지원 센터 — 계정과 사용 중 생기는 문제를 찾아보는 공식 지원 문서.