

초보자용 실습 교안

바이브코딩으로 웹 보드게임 만들기

초등 저학년 교육용 보드게임 사례로 배우는 AI 협업형 웹 프로그램 제작 입문

대상

바이브코딩을 처음 접하는 초보자

결과물

HTML 웹 보드게임 프
로토타입

핵심

말로 기획하고, AI와 수
정하며 완성하기



사례: 배움마을 대탐험 - 영어·수학·국어 융합 웹 보드게임

1. 이 교안의 목표

수업 후 할 수 있는 일

- 바이브코딩이 무엇인지 자신의 말로 설명할 수 있다.
- AI에게 웹 프로그램 제작을 요청하는 기본 프롬프트를 작성할 수 있다.
- HTML 파일을 받아 실행하고, 문제점을 찾아 수정 요청할 수 있다.
- 기획 → 제작 → 테스트 → 수정 → 재검증의 반복 과정을 이해한다.

핵심 메시지

코드를 먼저 외우는 수업이 아닙니다. “내가 만들고 싶은 결과물”을 명확히 말하고, AI와 대화하며 실제 작동하는 프로그램으로 발전시키는 방법을 배우는 수업입니다.

2. 바이브코딩이란?

바이브코딩은 사용자가 원하는 서비스의 목적, 분위기, 기능, 사용 장면을 자연어로 설명하고, AI가 그 의도를 코드로 구현하도록 협업하는 제작 방식입니다.

전통적 코딩 학습

문법 → 예제 → 코드 작성 → 실행

바이브코딩 학습

아이디어 → 프롬프트 → 실행
결과 → 피드백 → 개선

중요한 것은 “코드를 전혀 몰라도 된다”가 아니라, **코드를 모르더라도 시작할 수 있고, 결과물을 보며 점점 이해할 수 있다**는 점입니다.

3. 이번 수업의 완성 사례

이번 교안에서는 “초등학교 저학년 학생들이 영어·수학·국어를 재미있게 배울 수 있는 주사위 이동형 웹 보드게임”을 사례로 사용합니다.



레트로 픽셀 보드판



8개 이동 말 캐릭터와 게임 아이콘

구성 요소	웹게임에 반영된 기능
보드판	40칸 이동형 보드, 영어마을·수학숲·국어성·창의놀이터 등 구역 구성
참가자	2~8명 설정, 캐릭터 말 선택
학습 카드	영어·수학·국어·창의 미션 문제은행
난이도	학년 기준과 쉬움·보통·도전·혼합 난이도 설정
흥미 요소	레트로 이미지, 주사위, 별 토큰, 효과음, BGM
검증 요소	중복 질문 최소화, 안내 버튼, 승리 조건, 진행 로그

4. 제작 흐름 한눈에 보기

1

아이디어 말하기

2

규칙과 기능 정리

3

HTML 초안 생성

4

실행해 보기

5

부족한 점 말하기

6

이미지·사운드·난이도
추가

7

검증 후 배포

초보자는 처음부터 완벽한 요청을 하려고 하지 않아도 됩니다. 실행해 보고 “무엇이 불편한지”를 말하는 것이 바이트코딩의 핵심입니다.

5. 좋은 프롬프트의 기본 공식

목적 + 대상 + 결과물 형태 + 필수 기능 + 분위기 + 검증 요청

항목	질문	예시
목적	왜 만들까?	초등 저학년이 영어·수학·국어를 재미있게 배우게 하려고
대상	누가 쓸까?	1~3학년, 2~8명, 교실·방과후·가정 학습
형태	무엇으로 만들까?	HTML 하나로 실행되는 웹 보드게임
기능	무엇이 작동해야 할까?	참가자 설정, 주사위, 이동, 문제 카드, 점수, 승리 조건
분위기	어떤 느낌이면 좋을까?	레트로 픽셀, 귀여운 캐릭터, 효과음과 BGM
검증	무엇을 확인할까?	중복 문제 최소화, 버튼 작동, 난이도 반영, 단일 파일 실행

6. 그대로 따라 하는 1차 프롬프트

처음에는 아래 문장을 복사해 사용해도 됩니다. 괄호 안의 내용만 내 주제로 바꾸면 됩니다.

나는 [초등 저학년]이 사용할 수 있는 [교육용 웹 보드게임]을 만들고 싶어.
사용 인원은 [2~8명]이고, 주사위를 던져 보드판을 이동하는 방식이야.
게임 안에서 [영어, 수학, 국어]를 쉽고 재미있게 배울 수 있게 해줘.
HTML 파일 하나로 실행될 수 있게 만들고,
참가자 설정, 주사위, 이동 말, 문제 카드, 점수, 우승 조건을 포함해줘.
초보자도 사용할 수 있도록 화면 구성과 안내를 쉽게 만들어줘.

실습 위 프롬프트에서 교육 주제를 바꿔 보세요. 예: 환경교육, 안전교육, 진로교육, 디지털 리터러시, AI 리터러시

7. AI 결과물을 받은 뒤 반드시 확인할 것

확인 항목	질문	수정 요청 예시
실행 여부	파일을 열었을 때 화면이 보이냐?	버튼을 눌러도 시작이 안 돼. 시작 함수와 이벤트 연결을 점검해서 수정해줘.
사용자 흐름	초보자가 어디를 눌러야 할지 알 수 있냐?	첫 화면에 게임 안내 버튼을 만들고, 진행 방법을 쉽게 설명해줘.
반복 문제	질문이 너무 자주 반복되냐?	문제은행을 확장하고 이미 나온 질문은 최대한 다시 나오지 않게 해줘.
난이도	대상 연령에 맞냐?	1학년·2학년·3학년별로 문제 난이도가 달라지게 해줘.
흥미 요소	아이들이 재미있어할 요소가 있냐?	레트로 캐릭터, 효과음, 별 토큰 애니메이션을 추가해줘.
검증	수정 후 문제가 생기지 않았냐?	HTML 구조와 자바스크립트 오류를 점검하고, 작동 확인 내용을 알려줘.

8. 수정 요청은 이렇게 한다

바이브코딩에서 가장 중요한 능력은 “잘 안 된다”를 구체적으로 설명하는 것입니다.

아쉬운 요청

이거 이상해. 다시 해줘.

아쉬운 요청

더 재미있게 해줘.

좋은 요청

게임 시작 후 주사위 버튼은 보이지만, 클릭해도 말이 이동하지 않아. 주사위 클릭 이벤트와 말 위치 업데이트 함수를 확인해서 수정해줘.

좋은 요청

초등 저학년이 흥미를 느끼도록 레트로 픽셀 느낌의 캐릭터 말, 별 토큰 보상, 효과음, BGM 켜기 버튼을 추가해줘.

9. 이번 보드게임 제작 대화의 실제 흐름

1차 요청

초등 저학년용 영어·수학·국어 보드게임을 만들어 달라고 요청

기획안 생성

게임명, 규칙, 카드, 승리 조건, 수업 운영 방식 설계

웹 프로그램 요청

HTML 기반 웹 보드게임으로 구현 요청

흥미 요소 보완

레트로 보드판, 캐릭터, 효과음, BGM, 난이도 설정 추가 요청

중복 문제 개선

문제은행 확장과 중복 최소화 로직 추가 요청

교육 교안화

제작 경험을 바이브코딩 입문 강의 자료로 재구성

이 흐름 자체가 바이브코딩입니다. 한 번에 완성하는 것이 아니라, 결과물을 보면서 계속 더 구체적으로 요구하고 개선합니다.

10. 집에서 따라 해보기: 60분 실습 코스

시간	활동	해야 할 일
0~10분	아이디어 정하기	내가 만들고 싶은 교육 게임 주제 1개를 정한다.
10~20분	기본 프롬프트 작성	대상, 목적, 기능, 결과물 형태를 적는다.
20~35분	HTML 초안 만들기	AI에게 단일 HTML 파일로 만들어 달라고 요청한다.
35~45분	실행과 오류 확인	파일을 열어 버튼, 화면, 점수, 이동이 되는지 확인한다.
45~55분	수정 요청	불편한 점을 구체적으로 말하고 개선을 요청한다.
55~60분	저장과 공유	파일명을 정리하고, 다음에 추가할 기능을 메모한다.

11. 초보자용 파일 실행 방법

1. AI가 만들어 준 HTML 파일을 다운로드합니다.
2. 파일 이름이 너무 길면 짧게 바꿉니다. 예: `my_game.html`
3. 파일을 더블클릭하면 크롬, 엣지 같은 웹브라우저에서 열립니다.
4. 버튼을 눌러 기능이 작동하는지 확인합니다.
5. 작동하지 않는 부분은 화면을 캡처하거나 증상을 적어 AI에게 수정 요청합니다.

주의 BGM은 브라우저 보안 정책 때문에 자동으로 시작되지 않을 수 있습니다. 사용자가 “BGM 켜기” 버튼을 눌러야 정상 재생되는 경우가 많습니다.

12. 프롬프트 템플릿 모음

기본 제작 요청

[대상]을 위한 [주제] 웹 프로그램을 만들고 싶어.
HTML 하나로 실행되게 만들고, 초보자가 쓰기 쉽게 화면과 버튼을 구성해줘.
필수 기능은 [기능 1], [기능 2], [기능 3]이야.
디자인은 [분위기] 느낌으로 해줘.

수정 요청

현재 결과물에서 [문제 상황]이 발생해.
원인은 [추정 원인]일 수 있으니 확인해주고,
[원하는 결과]가 되도록 수정해줘.
수정 후 어떤 부분을 바꿨는지도 간단히 설명해줘.

검증 요청

완성된 HTML 파일을 검증해줘.
확인할 항목은 버튼 작동, 화면 전환, 데이터 중복, 난이도 반영,
오류 메시지 발생 여부, 단일 파일 실행 가능 여부야.

고급화 요청

현재 웹 프로그램을 교육 현장에서 더 흥미롭게 사용할 수 있도록
캐릭터, 효과음, 안내 팝업, 난이도 선택, 문제은행 확장,
수업 운영 팁을 추가해서 업그레이드해줘.

13. 실습 워크시트

1) 내가 만들고 싶은 프로그램 주제

2) 사용할 사람 예: 초등학생, 시니어, 강사, 학부모

3) 꼭 필요한 기능 3가지

4) 원하는 분위기 예: 귀여운, 전문적인, 레트로, 카드뉴스 느낌

5) 첫 번째 프롬프트 작성

14. 자주 생기는 문제와 해결법

문제	가능한 원인	해결 요청 문장
버튼이 눌리지 않음	이벤트 연결 누락	버튼 클릭 이벤트가 연결되어 있는지 확인하고 수정해줘.
화면이 깨짐	CSS 레이아웃 문제	모바일과 PC에서 모두 보기 좋게 반응형 레이아웃으로 수정해줘.
질문이 반복됨	문제은행 부족 또는 랜덤 로직 단순	문제은행을 확장하고 최근 출제 문제를 제외하는 로직을 추가해 줘.
난이도가 안 맞 음	문제 분류 부족	학년별·난이도별 문제를 분리하고 선택값에 따라 출제되게 해줘.
소리가 안 남	브라우저 자동재생 제한	사용자가 버튼을 눌렀을 때만 BGM이 시작되도록 수정해줘.
파일이 너무 큼	이미지 내장 용량 증가	이미지를 압축하거나 외부 assets 폴더 방식으로 분리하는 버전을 만들어줘.

15. 수업 진행자용 운영안

구분	내용
도입 10분	바이브코딩 개념 소개, 완성 예시 시연
전개 30분	보드게임 제작 대화 흐름 설명, 프롬프트 공식 연습
실습 40분	학습자별 웹게임 또는 간단한 웹도구 만들기
공유 20분	서로 결과물 실행, 잘 된 점과 수정할 점 나누기
마무리 10분	집에서 해볼 과제 안내, 다음 단계 제시

강의 팁

초보자 수업에서는 코드 설명을 길게 하기보다 “어떤 요청을 하면 결과가 어떻게 바뀌는지”를 보여주는 것이 효과적입니다.

16. 완성도 체크리스트

☐ 대상자가 누구인지 명확하다.

☐ 프로그램의 목적이 한 문장으로 설명된다.

☐ HTML 파일이 정상 실행된다.

☐ 첫 화면에서 무엇을 해야 하는지 알 수 있다.

☐ 버튼과 입력창이 정상 작동한다.

☐ 오류가 생겼을 때 수정 요청을 할 수 있다.

☐ 디자인 분위기가 대상자에게 적절하다.

☐ 문제은행이나 콘텐츠가 충분하다.

☐ 안내 버튼 또는 사용 설명이 있다.

☐ 다음 업그레이드 방향을 정리했다.

17. 마무리

바이브코딩은 코딩을 몰라도 “내가 만들고 싶은 것”을 실제 프로그램으로 바꾸는 강력한 출발점입니다. 하지만 좋은 결과물을 얻으려면 사용자의 역할도 중요합니다.

- 처음 요청은 넓게 하되, 결과를 보고 구체적으로 수정합니다.
- “더 좋게”보다 “어떤 기능을 어떻게 바꿀지”를 말합니다.
- 실행해 보고 검증하는 과정을 반드시 포함합니다.
- 완성품보다 반복 개선 경험이 더 중요합니다.

오늘의 핵심: 아이디어를 말하고, 결과를 보고, 다시 요청하면 프로그램은 점점 좋아집니다.