

12주 단기 완성: "AI-Powered MVP 마스터 클래스"

타겟: 비전공자, 창업 희망자, 혹은 빠르게 포트폴리오 1개가 필요한 주니어 개발자로 취업 준비하는 준비생 목표: 복잡한 이론 최소화, 일단 돌아가는 서비스를 AI와 함께 가장 빠르게 런칭하기

주차	단계	주제	핵심 내용 (AI 활용 포인트)
1-2	기획 & 기초	AI와 셋업하기	<ul style="list-style-type: none">- Cursor/VS Code, ChatGPT 세팅- AI에게 요구사항 정의서 쓰게 하기 (기획의 구체화)- HTML/CSS 구조 잡기 (AI가 생성하고 사람은 '배치'만 수정)
3-4	Frontend	React + AI	<ul style="list-style-type: none">- "Vibe Coding" 입문: "로그인 화면 만들어줘" 대신 "모던한 디자인의 로그인 컴포넌트 짜줘"- 컴포넌트 구조 이해 (AI 코드를 읽는 법 교육)

			<ul style="list-style-type: none"> - Tailwind CSS로 AI 디자인 즉시 수정하기
5-6	Logic	Javascript Essentials	<ul style="list-style-type: none"> - AI가 못 잡는 버그 잡기: 변수 스코프, 비동기(Async/Await) 등 AI가 실수하는 포인트만 골라 교육 - "이 코드가 왜 어려인지 AI에게 물어보고 해결하는 법" 실습
7-8	Backend	Serverless & API	<ul style="list-style-type: none"> - Node.js/Express 기초 - AI로 API 명세서 작성 및 코드 생성 - Firebase나 Supabase 연동 (AI가 복잡한 백엔드 설정 없이 DB 연결하도록 유도)
9-10	Full Stack	기능 통합	<ul style="list-style-type: none"> - 프론트와 백엔드 연결 (Axios/Fetch) - AI를 이용한 데이터 더미(Mock Data) 생성 및 테스트

			<ul style="list-style-type: none"> - CRUD 구현 (AI에게 "게시판 CRUD 로직 짜줘" 하고 검수)
11-1 2	Deploy	배포 & 런칭	<ul style="list-style-type: none"> - Vercel/Netlify 원클릭 배포 - 도메인 연결 - 최종 "바이브" 튜닝: UI/UX 디테일을 자연어로 명령하며 수정 (Cursor 활용)

20주 정규 과정: “Future Full-Stack 아키텍트 과정”

타겟: 주니어 개발자 신규 취업뿐만 아니라 중간 레벨의 개발자 전직을 원하시는 분. AI가 있어도 대체되지 않을 개발자를 꿈꾸는 사람 목표: AI를 도구로 쓰되, AI가 해결 못 하는 깊은 문제(데이터 설계, 보안, 최적화)를 해결하는 능력 배양

주차	단계	주제	핵심 내용 (AI 활용 + Deep Dive)
1-4	Foundation	언어의 본질	<ul style="list-style-type: none"> - JS Deep Dive (실행 컨텍스트, 클로저) - "AI는 거짓말을 한다": AI가 짠 코드의 논리적 오류 찾아내기 훈련

			<ul style="list-style-type: none"> - 자료구조/알고리즘 기초 (AI 코드를 최적화하기 위해 필요)
5-8	Frontend	React Deep Dive	<ul style="list-style-type: none"> - 상태 관리(Redux/Zustand)와 헥(Hooks)의 원리 - 컴포넌트 설계 철학: AI에게 "재사용 가능한 버튼 컴포넌트"를 시키기 위한 프롬프트 엔지니어링 - 퍼포먼스 최적화 (Rendering 이슈 해결)
9-12	Database	데이터 설계	<ul style="list-style-type: none"> - 이건 인간이 해야 함: ERD 설계 및 정규화 (AI는 비즈니스 로직을 모름) - SQL vs NoSQL 선택 기준 - AI를 활용한 복잡한 쿼리 작성 및 튜닝

13-1 6	Backend	API & Auth	<ul style="list-style-type: none"> - RESTful API vs GraphQL - JWT, OAuth 인증 구현 (보안은 AI에게 맡기면 위험, 원리 파악 필수) - 에러 핸들링 및 로깅 전략 (실무 노하우 전수)
17-2 0	Capstone	실전 프로젝트	<ul style="list-style-type: none"> - CI/CD 파이프라인 구축 (GitHub Actions with AI) - 클라우드 배포 (AWS/GCP 맛보기) - 코드 리팩토링: AI가 짠 '돌아만 가는 코드'를 '유지보수 좋은 코드'로 다듬기

- 주차 별 상세 진행은 진행 속도와 다른 추가 요구 사항에 따라 조금씩 바뀔 수 있습니다.