

목차

프롤로그: 두 사람의 이야기

부모님 집에서 시작한 청년

27번 실패한 개발자

그들과 당신의 차이

이 책이 도와드릴 것

시작하기 전에

Chapter 1. AI 시대, 코딩의 새로운 패러다임

학습 목표

1. 전통적인 코딩 vs 바이브 코딩

2. AI가 코딩을 어떻게 바꾸고 있는가

3. 왜 지금이 시작하기 좋은 시기인가

실생활 비유: 요리로 이해하는 바이브 코딩

실제 사례: Pieter Levels 이야기

핵심 포인트

다음 챕터 미리보기

Chapter 2. 바이브 코딩 vs 전통적 코딩

학습 목표

1. 전통적 코딩의 어려움

2. 바이브 코딩의 장점

실생활 비유: 자동차 운전 vs 자율주행

실제 사례: Marc Lou의 27번의 시도

핵심 포인트

다음 챕터 미리보기

Chapter 3. 당신도 할 수 있는 이유

학습 목표

1. 나이는 숫자에 불과합니다

2. 경험이 곧 아이디어

3. 작게 시작하는 용기

실생활 비유: 씨앗 심기

실제 사례: 늦게 시작해도 성공할 수 있습니다

핵심 포인트

다음 챕터 미리보기

Part 1 마무리: 바이브 코딩 마인드셋 점검

Part 1 핵심 요약

실습 과제: 내가 만들고 싶은 것 3가지 적어보기

자가 진단 체크리스트

당신도 할 수 있습니다

Part 2 미리보기: 본격적으로 시작하기

Chapter 4. AI 도구 설치하고 첫 대화 나누기

학습 목표

1. Claude 가입하기

2. Cursor 설치하기

3. AI와 첫 대화 나누기

실생활 비유: 새 스마트폰 처음 켜기

실제 사례: 처음 AI를 써본 사람들의 반응

핵심 포인트

다음 챕터 미리보기

Chapter 5. 10분 만에 첫 웹페이지 만들기

학습 목표

1. AI에게 웹페이지 요청하기

2. 수정 요청하기

3. 더 멋진 웹페이지 만들기

실생활 비유: 레고 블록으로 집 만들기

실제 사례: 바이브 코딩 10분 챌린지

핵심 포인트

다음 챕터 미리보기

Chapter 06. 내 웹페이지를 세상에 공개하기

학습 목표

1. 배포란 무엇인가요?

2. Netlify로 배포하기

3. 도메인 이야기

실생활 비유: 가게 문 열기

실제 사례: Pieter Levels의 첫 배포 이야기

핵심 포인트

다음 챕터 미리보기

Part 2 마무리: 첫 웹사이트 만들기 프로젝트

Part 2 핵심 요약

실습 과제: 나만의 자기소개 웹페이지를 만들고 배포하기

자가 진단 체크리스트

당신도 할 수 있습니다

Part 3 미리보기: 본격 프로젝트 시작하기

Chapter 07. 실전 프로젝트 기획하기

학습 목표

1. 아이디어는 불편함에서 시작합니다

2. MVP: 핵심 기능만 담은 첫 버전

실생활 비유: 식당 메뉴판

실제 사례: Marc Lou의 ShipFast

3. 내 프로젝트 기획하기

핵심 포인트

다음 챕터 미리보기

Chapter 08. 기능 하나씩 추가하기

학습 목표

1. 좋은 프롬프트 작성법

2. 기능 추가 순서

3. 에러 해결하기

실생활 비유: 집 인테리어

실제 사례: NomadList 기능 추가 히스토리

핵심 포인트

다음 챕터 미리보기

Chapter 09. 디자인 다듬기

학습 목표

1. 색상과 폰트 바꾸기
2. 레이아웃 개선하기
3. 모바일 대응 (반응형 디자인)

실생활 비유: 옷 입히기

실제 사례: 심플한 디자인으로 성공한 서비스들

핵심 포인트

다음 챕터 미리보기

Chapter 10. 사용자 피드백 받고 개선하기

학습 목표

1. 주변 5명에게 보여주기
2. 피드백 정리하기: 중요/나중에/무시
3. 우선순위 정해서 개선하기

실생활 비유: 시식 코너

실제 사례: Pieter Levels의 트위터 피드백 전략

핵심 포인트

다음 챕터 미리보기

Part 3 마무리: 나만의 앱 아이디어 구체화

Part 3 핵심 요약

실습 과제: 내 프로젝트 기획서 1장 작성하기

자가 진단 체크리스트

당신도 할 수 있습니다

Part 4 미리보기: 수익화 시작하기

Chapter 11. 수익화 전략 세우기

학습 목표

1. 수익 모델의 종류

2. 내 프로젝트에 맞는 모델 찾기

3. 가격 정하기

실생활 비유: 과일 가게

실제 사례: Pieter Levels의 수익 모델

핵심 포인트

다음 챕터 미리보기

Chapter 12. 첫 수익 만들기

학습 목표

1. 결제 시스템 연동하기

2. 첫 고객 찾기

3. 가격 테스트하기

실생활 비유: 시장에서 첫 손님 맞이하기

실제 사례: Marc Lou의 첫 100달러

핵심 포인트

다음 챕터 미리보기

Chapter 13. 마케팅의 기초

학습 목표

1. SNS 활용하기

2. 콘텐츠 마케팅

3. 커뮤니티 참여하기

실생활 비유: 동네 입소문

실제 사례: Marc Lou의 트위터 마케팅

핵심 포인트

다음 챕터 미리보기

Chapter 14. 수익 확장하기

학습 목표

1. 두 번째 제품 만들기

2. 기존 제품 업그레이드

실생활 비유: 과수원 키우기

실제 사례: Pieter Levels의 다중 프로젝트 전략

핵심 포인트

다음 챕터 미리보기

Part 4 마무리: 수익화 계획 세우기

Part 4 핵심 요약

실습 과제: 내 프로젝트의 수익 모델 캔버스 작성하기

자가 진단 체크리스트

당신도 할 수 있습니다

Part 5 미리보기: 지속 가능한 성장

Chapter 15. 자동화로 시간 벌기

학습 목표

1. 반복 작업 자동화하기
2. AI 챗봇으로 고객 응대하기
3. 모니터링 자동 설정

실생활 비유: 세탁기와 식기세척기

실제 사례: 1인 운영자의 하루

핵심 포인트

다음 챕터 미리보기

Chapter 16. 나만의 커뮤니티 만들기

학습 목표

1. 사용자 커뮤니티의 힘
2. 커뮤니티 시작하기
3. 커뮤니티 운영 노하우

실생활 비유: 동네 모임

실제 사례: NomadList 커뮤니티

핵심 포인트

다음 챕터 미리보기

Chapter 17. 1인 사업가의 장기 비전

학습 목표

1. 지속 가능한 사업

2. 건강과 균형

3. AI 기술의 미래와 기회

실생활 비유: 마라톤

실제 사례: Pieter Levels 10년의 여정

핵심 포인트

마무리

Part 5 마무리: 지속 가능한 1인 사업 로드맵

Part 5 핵심 요약

실습 과제: 나의 12개월 로드맵 만들기

자가 진단 체크리스트

당신도 할 수 있습니다

에필로그 안내

에필로그: 당신의 이야기는 이제 시작입니다

여기까지 오신 당신에게

여러분이 얻은 것들

한 가지 부탁

완벽하지 않아도 시작하세요

Pieter Levels가 당신에게 하고 싶은 말

부록

A. 유용한 도구 목록

B. 자주 묻는 질문 (FAQ)

C. 용어 사전

D. 이 책에서 소개한 인물

바이브 코딩으로 1인 사업가 되기

코딩 몰라도 괜찮아요, AI가 함께 합니다



Author: J.M

프롤로그: 두 사람의 이야기

부모님 집에서 시작한 청년

2014년, 네덜란드의 한 청년이 있었습니다.

이름은 Pieter Levels(피터 레벨스).

그는 부모님 집에서 지내고 있었습니다.

우울증을 겪고 있었습니다.

특별한 학위도, 직장도 없었습니다.

그가 한 일은 단순했습니다.

“12개월 안에 12개 프로젝트를 만들자.”

노트북 하나로 시작했습니다.

대단한 기술이 있던 것은 아닙니다.

간단한 코딩을 독학으로 배웠을 뿐입니다.

12개 중 대부분은 실패했습니다.

하지만 몇 개가 살아남았습니다.

그중 하나가 NomadList(노매드리스트)입니다.

디지털 노마드를 위한 도시 정보 사이트입니다.

2025년 현재, 그의 연 수입은 약 38억 원입니다.

직원 없이 혼자서 운영합니다.

27번 실패한 개발자

또 한 사람이 있습니다.

Marc Lou(마크 루).

그는 프랑스 출신의 1인 개발자입니다.

그에게도 화려한 시작은 없었습니다.

만든 제품이 27번이나 실패했습니다.

27번이라는 숫자를 생각해 보세요.

보통 사람이라면 5번쯤에서 포기했을 것입니다.

하지만 그는 계속했습니다.

전략을 바꿨습니다.

“작은 제품을 빠르게 만들고,

반응이 있는 것만 키우자.”

28번째 시도가 성공했습니다.

그리고 29번째, 30번째도 성공했습니다.

지금 그는 월 수천만 원을 벌고 있습니다.

역시 혼자서 운영합니다.

그들과 당신의 차이

이 두 사람의 이야기를 들으면 이런 생각이 드실 수 있습니다.

“그 사람들은 특별한 거 아니야?”

“젊으니까 가능했던 거 아닌가?”

“나는 코딩을 모르는데…”

한 가지 중요한 사실이 있습니다.

Pieter와 Marc가 시작할 때는 AI가 없었습니다.

그들은 코딩을 직접 배워야 했습니다.
에러가 나면 혼자 해결해야 했습니다.
모든 것을 손으로 만들어야 했습니다.

2025년 지금, 여러분에게는 AI가 있습니다.

- 코드를 몰라도 AI가 만들어 줍니다
- 에러가 나면 AI가 고쳐 줍니다
- 디자인도 AI가 도와줍니다
- 마케팅 문구도 AI가 써줍니다

그들보다 훨씬 유리한 조건에서 시작하는 것입니다.

이 책이 도와드릴 것

이 책은 여러분을 위해 쓰였습니다.

코딩을 전혀 모르는 분.
컴퓨터가 조금 어려운 분.
하지만 **“나도 뭔가 만들어 보고 싶다”**는 마음이 있는 분.

이 책을 다 읽고 나면 이런 것들을 할 수 있게 됩니다.

1. AI 도구를 자유롭게 사용할 수 있습니다
2. 나만의 웹사이트나 서비스를 만들 수 있습니다
3. 만든 것을 인터넷에 공개할 수 있습니다
4. 첫 수익을 만드는 방법을 알게 됩니다
5. 지속적으로 성장하는 방법을 배웁니다

어렵지 않습니다.
천천히, 한 걸음씩 함께 가겠습니다.

시작하기 전에

한 가지만 약속해 주세요.

“완벽하지 않아도 괜찮다.”

이 책을 읽으면서 막히는 부분이 있을 수 있습니다.

이해가 안 되는 부분도 있을 수 있습니다.

괜찮습니다.

다시 읽으면 됩니다.

AI에게 물어보면 됩니다.

주변에 도움을 요청하면 됩니다.

중요한 건 **멈추지 않는 것**입니다.

Pieter Levels도 처음에는 아무것도 몰랐습니다.

Marc Lou도 27번이나 실패했습니다.

그들이 성공한 비결은 단 하나입니다.

멈추지 않았다는 것.

자, 이제 시작해 볼까요?

첫 페이지를 넘겨 주세요.

Chapter 1. AI 시대, 코딩의 새로운 패러다임

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. 전통적인 코딩과 바이브 코딩이 어떻게 다른지 이해합니다.
 2. AI가 코딩을 어떻게 바꾸고 있는지 큰 그림을 그릴 수 있습니다.
 3. 지금이 시작하기에 가장 좋은 시기인 이유를 설명할 수 있습니다.
-

1. 전통적인 코딩 vs 바이브 코딩

코딩, 꼭 어렵게 배워야 할까?

“코딩을 배우려면 영어부터 해야 하나요?”

많은 분들이 이렇게 물어봅니다.

솔직히 말하면, 예전에는 맞는 말이었습니다.

전통적인 코딩은 이런 과정이었습니다.

프로그래밍 언어를 배운다 →
문법을 외운다 →
에러를 찾는다 →
또 에러를 찾는다 →
몇 달이 지나서야 뭔가 만들어진다

대학교에서 4년을 배워도 막상 실무에서는 또 새로 배워야 했습니다.

바이브 코딩은 다릅니다

바이브 코딩은 완전히 다른 방식입니다.

내가 원하는 것을 말로 설명하면, AI가 코드를 만들어 줍니다.

예를 들어 볼까요?

전통적인 코딩	바이브 코딩
HTML, CSS, JavaScript를 배운다	"예약 페이지 만들어줘"라고 말한다
코드를 한 줄 한 줄 직접 작성한다	AI가 코드를 만들어 준다
에러가 나면 직접 찾아 고친다	AI에게 "에러 고쳐줘"라고 말한다
완성까지 몇 주~몇 달	완성까지 몇 시간~며칠

전통 코딩 vs 바이브 코딩

전통 코딩



- 수개월 학습 필요



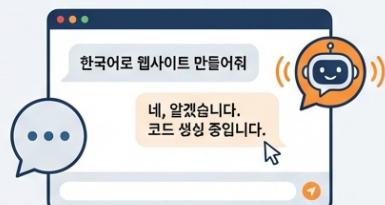
- 프로그래밍 언어 암기



- 에러 직접 디버깅



바이브 코딩



- 바로 시작 가능



- 한국어로 대화



- AI가 에러 해결



바이브 코딩에서 가장 중요한 능력은 이것입니다.

“내가 뭘 만들고 싶은지 정확히 설명하는 능력”

코드를 몰라도 됩니다.

영어를 잘 몰라도 됩니다.

한국어로 설명하면 됩니다.

2. AI가 코딩을 어떻게 바꾸고 있는가

10년 전과 지금의 차이

2015년에 간단한 웹사이트를 만들려면 이런 것들이 필요했습니다.

- HTML/CSS 기초 학습 (2~4주)
- JavaScript 기본 문법 (4~8주)
- 서버 구축 방법 (4~8주)
- 데이터베이스 설계 (2~4주)

최소 3개월은 공부해야 겨우 시작할 수 있었습니다.

2025년, 지금은 어떨까요?

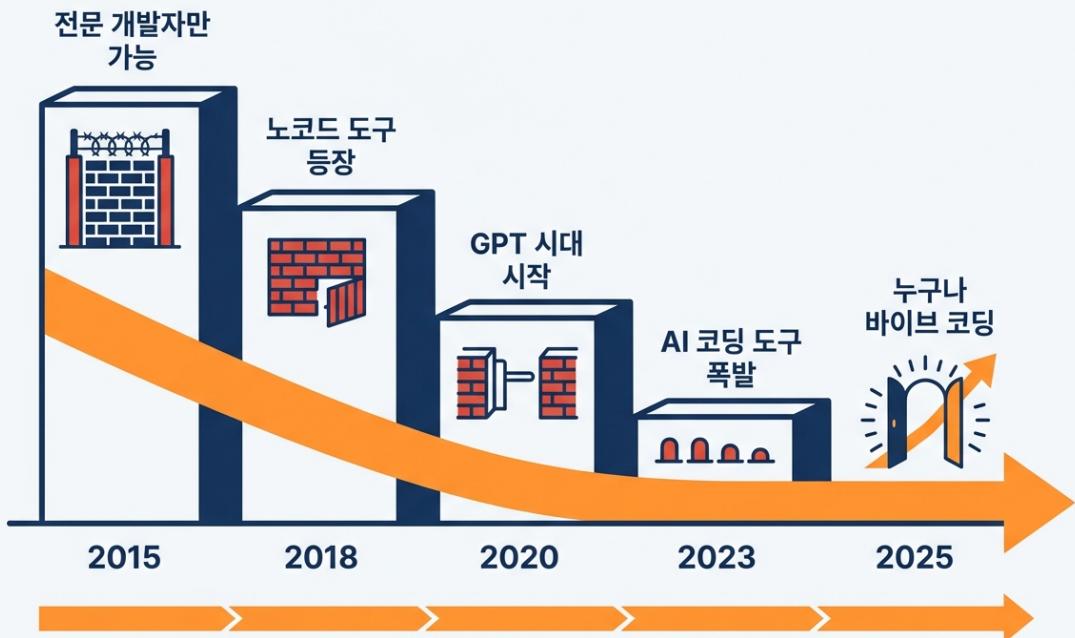
지금은 AI 도구 하나만 있으면 됩니다.

- [Claude](#) — 코드를 대신 작성해 줍니다
- [Cursor](#) — AI가 내장된 코드 편집기입니다
- [Bolt/Lovable](#) — 말로 설명하면 웹사이트가 만들어집니다
- [Vercel/Netlify](#) — 클릭 몇 번으로 전 세계에 공개됩니다

이 도구들은 대부분 [무료](#)이거나 월 2~3만원이면 충분합니다.

진입 장벽이 사라졌습니다

코딩 진입장벽의 변화 (2015→2025)



예전에는 이런 사람만 개발자가 될 수 있었습니다.

“컴퓨터공학과 졸업”

“코딩 학원 6개월 수료”

“개발 경력 3년 이상”

지금은 다릅니다.

“내가 이런 서비스를 만들고 싶어”

이 한 마디면 시작할 수 있습니다.

3. 왜 지금이 시작하기 좋은 시기인가

세 가지 이유

첫째, AI 도구가 충분히 성숙했습니다.

2022년 ChatGPT가 처음 나왔을 때는 아직 부족했습니다.
코드를 만들어 달라고 하면 절반은 틀렸습니다.

2025년 지금은 다릅니다.

AI가 만드는 코드의 품질이 크게 좋아졌습니다.
간단한 서비스라면 거의 완벽하게 만들어 냅니다.

둘째, 성공 사례가 쌓이고 있습니다.

바이브 코딩으로 실제 돈을 버는 사람들이 나타났습니다.
이 책에서도 여러 사례를 소개하겠습니다.
"정말 되는 거구나"를 직접 확인할 수 있습니다.

셋째, 경쟁자가 아직 적습니다.

대부분의 사람들은 아직 이렇게 생각합니다.

“코딩? 나는 그런 거 못 해.”
“AI? 짧은 사람들이나 쓰는 거 아냐?”

그래서 지금이 기회입니다.

남들이 망설이는 사이에 시작하면 됩니다.

실생활 비유: 요리로 이해하는 바이브 코딩

코딩이 아직 어렵게 느껴지시나요?

요리에 비유해 보겠습니다.

전통적인 코딩 = 재료부터 직접 만들어 요리하기

밀가루를 직접 빻고, 면을 직접 뽑고, 소스를 직접 만듭니다.
맛있는 파스타 한 접시를 만들려면 몇 시간이 걸립니다.
전문 요리사만 할 수 있는 일이었습니다.

바이브 코딩 = 밀키트로 요리하기

밀키트를 주문하면 재료가 손질되어 옵니다.
레시피대로 따라하면 30분이면 완성됩니다.
요리 초보도 맛있는 한끼를 만들 수 있습니다.

AI 도구 = 똑똑한 요리 보조 셰프

옆에서 전문 셰프가 도와준다고 상상해 보세요.

“지금 불 세기를 좀 줄이세요.”
“소금은 이 정도만 넣으면 됩니다.”
“이 순서로 하면 더 맛있어요.”

AI 도구가 바로 이 역할을 합니다.
내가 방향을 정하면 AI가 실행을 도와줍니다.
중요한 건 요리사(나)의 아이디어입니다.
무엇을 만들지 결정하는 건 바로 나입니다.

실제 사례: Pieter Levels 이야기

부모님 집에서 시작한 1인 사업가

Pieter Levels(피터 레벨스)라는 사람이 있습니다.
네덜란드 출신의 1인 개발자입니다.

2014년, 그는 이런 상황이었습니다.

- 부모님 집에 얹혀살고 있었습니다
- 우울증을 겪고 있었습니다
- 특별한 기술이나 학위가 없었습니다
- 돈도 거의 없었습니다

그가 한 일은 단순했습니다.

“사람들에게 필요한 간단한 웹사이트를 만들자.”

12개 프로젝트 도전

그는 "12개월 안에 12개 프로젝트 만들기"에 도전했습니다.

대단한 기술이 필요한 게 아니었습니다.

간단한 도구를 사용해서 빠르게 만들었습니다.

실패한 것도 많았지만, 몇 개가 성공했습니다.

그중 하나가 [NomadList](#)(노매드리스트)입니다.

디지털 노마드를 위한 도시 정보 사이트입니다.

그리고 지금

2025년 현재, Pieter Levels의 연 수입은 약 38억 원입니다.

직원 없이 혼자서 운영합니다.

여기서 중요한 포인트가 있습니다.

2014년에는 간단한 사이트를 만드는 것도 꽤 어려웠습니다.

그때도 기본적인 코딩 지식은 필요했습니다.

2025년 지금은 AI가 있습니다.

그때보다 훨씬 더 쉽게 시작할 수 있습니다.

Pieter Levels도 최근 인터뷰에서 이렇게 말했습니다.

“지금 시작하는 사람들이 부럽다.
AI 덕분에 예전보다 10배는 빠르게 만들 수 있다.”

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. **바이브 코딩**은 코드를 직접 작성하는 것이 아니라, AI에게 원하는 것을 설명하는 방식입니다.
2. AI 도구가 충분히 발전해서, 코딩을 몰라도 서비스를 만들 수 있는 시대가 되었습니다.
3. 가장 중요한 능력은 프로그래밍이 아니라 **“무엇을 만들고 싶은지 아는 것”**입니다.
4. 지금이 시작하기 가장 좋은 시기입니다. AI는 충분히 좋아졌고, 경쟁자는 아직 적습니다.
5. Pieter Levels처럼 특별한 배경 없이도 시작할 수 있습니다. 지금은 그때보다 훨씬 쉽습니다.

다음 챕터 미리보기

Chapter 2에서는 바이브 코딩에 사용하는 **AI 도구들을** 하나씩 살펴봅니다.

- Claude, Cursor, Bolt 같은 도구가 뭔지
- 어떤 도구를 먼저 써보면 좋은지
- 무료로 시작하는 방법

도구를 알아야 요리를 시작할 수 있겠죠?

다음 챕터에서 여러분의 주방을 세팅해 보겠습니다.

Chapter 2. 바이브 코딩 vs 전통적 코딩

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. 전통적 코딩이 왜 어려웠는지 구체적으로 이해합니다.
 2. 바이브 코딩이 가진 [핵심 장점 3가지](#)를 설명할 수 있습니다.
 3. 실패 비용이 거의 없다는 사실에 자신감을 얻습니다.
-

1. 전통적 코딩의 어려움

외국어를 배우는 것과 같았습니다

전통적 코딩은 새로운 언어를 배우는 것이었습니다.

영어도 아닌, 컴퓨터만 알아듣는 언어였습니다.

```
function calculateTotal(items) {  
    return items.reduce((sum, item) => sum + item.price, 0);  
}
```

이 코드가 무슨 뜻인지 아시나요?

"물건들의 가격을 다 합쳐라"라는 뜻입니다.

이걸 쓰려면 몇 달을 배워야 했습니다.

에러와의 끝없는 싸움

코드를 작성하면 에러가 납니다.

점 하나, 쉼표 하나 빠져도 안 돌아갑니다.

에러 메시지는 영어로 나옵니다.

무슨 뜻인지 모르겠습니다.

검색해서 찾아봐야 합니다.

찾아봐도 이해가 안 됩니다.

이 과정에서 대부분 포기합니다.

배워야 할 것이 너무 많았습니다

웹사이트 하나를 만들려면 이것들을 알아야 했습니다.

- **HTML**: 웹페이지의 뼈대
- **CSS**: 웹페이지의 디자인
- **JavaScript**: 웹페이지의 동작
- **서버**: 데이터를 저장하는 곳
- **데이터베이스**: 정보를 정리하는 곳

이 다섯 가지를 다 배우려면 최소 6개월이 걸립니다.

6개월 뒤에도 초보입니다.

2. 바이브 코딩의 장점

말로 하면 됩니다

바이브 코딩에서는 코드를 쓸 필요가 없습니다.

전통적 코딩:

코드를 직접 작성한다

바이브 코딩:

"로그인 페이지를 만들어줘"라고 말한다

이것이 전부입니다.

AI가 코드를 대신 작성해 줍니다.

속도가 다릅니다

전통적 코딩으로 로그인 페이지를 만들면?

초보자 기준 2~3일이 걸립니다.

바이브 코딩으로 같은 페이지를 만들면?

10분이면 됩니다.

작업	전통적 코딩	바이브 코딩
로그인 페이지	2~3일	10분
게시판	1~2주	30분~1시간
결제 기능	3~5일	1시간
전체 웹사이트	1~3개월	1~3일

10배 이상 빠릅니다.

개발 속도 비교



실패 비용이 거의 제로입니다

전통적으로 서비스를 만들려면 이런 비용이 들었습니다.

- 개발자 고용: 월 수백만 원
- 외주 개발: 수천만 원
- 시간 투자: 최소 3~6개월

돈을 들이고 시간을 들여서 만들었는데 실패하면?

큰 손해입니다.

바이브 코딩은 다릅니다.

- AI 도구 비용: 무료 또는 월 2~3만 원
- 시간 투자: 며칠
- 실패하면? 다시 만들면 됩니다

실패해도 잊는 것이 거의 없습니다.

그래서 마음 편하게 도전할 수 있습니다.

실생활 비유: 자동차 운전 vs 자율주행

전통적 코딩 = 수동 변속기 운전 배우기

예전에는 운전을 배우려면 많은 것을 알아야 했습니다.

클러치를 밟고, 기어를 넣고, 악셀을 조절합니다.

타이밍이 안 맞으면 차가 덜컹거립니다.

시동이 꺼지기도 합니다.

운전면허를 따는 데만 몇 달이 걸렸습니다.

면허를 따도 도로에 나가면 무서웠습니다.

전통적 코딩이 이렇습니다.

배울 것이 많고, 실수가 잦고, 시간이 오래 걸립니다.

바이브 코딩 = 자율주행차 타기

자율주행차를 상상해 보세요.

“강남역으로 가줘.”

이 한마디면 됩니다.

클러치도 모르고, 기어도 모르고, 도로 법규도 몰라도 됩니다.

차가 알아서 갑니다.

바이브 코딩이 이렇습니다.

“로그인 페이지 만들어줘.”

이 한마디면 AI가 알아서 만들어 줍니다.

내가 해야 할 일은 목적지를 정하는 것뿐입니다.

중요한 건 어디로 갈지 아는 것

수동 운전이든 자율주행이든 공통점이 있습니다.

어디로 갈지는 내가 정해야 한다는 것입니다.

바이브 코딩도 마찬가지입니다.
무엇을 만들지 결정하는 건 나입니다.
AI는 그 결정을 실행해 주는 도구입니다.

실제 사례: Marc Lou의 27번의 시도

실패가 무서운 이유

“만들어 봤자 실패하면 어쩌지?”
이런 걱정이 가장 큰 장벽입니다.
특히 시간과 돈이 많이 들었을 때 실패가 무섭습니다.

Marc Lou는 27번 실패했습니다

Marc Lou(마크 루)라는 사람이 있습니다.
프랑스 출신의 1인 개발자입니다.
그는 무려 27개의 제품을 만들었습니다.
그리고 27개 모두 실패했습니다.
보통 사람이라면 5번쯤에서 포기했을 겁니다.
10번이면 “나는 안 되는 사람인가 보다” 생각했을 겁니다.
하지만 Marc Lou는 계속했습니다.
왜 가능했을까요?

실패 비용이 작았기 때문입니다

Marc Lou는 각 제품을 며칠 만에 만들었습니다.
비용도 거의 들지 않았습니다.
실패해도 잃는 것이 거의 없었습니다.
“안 되면 다음 거 만들자.”
이런 마음으로 가볍게 시도할 수 있었습니다.

전통적 코딩이었다면 27번 시도가 가능했을까요?
각각 3개월씩 걸렸다면 **7년**이 필요합니다.
현실적으로 불가능합니다.

28번째가 성공했습니다

27번 실패 후, 28번째 제품이 드디어 성공했습니다.
그리고 그 뒤로도 연달아 성공했습니다.

지금 Marc Lou는 월 수천만 원을 벌고 있습니다.
혼자서 운영합니다.

핵심 교훈은 이것입니다.

“빠르게 만들고, 빠르게 실패하고, 빠르게 다시 시작하라.”

바이브 코딩이기에 가능한 전략입니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. **완벽함보다 실행이 중요합니다.** 27번 실패한 Marc Lou가 증명했습니다.
2. **AI가 반복 작업을 대신합니다.** 코드 작성, 에러 수정, 디자인까지 AI가 도와줍니다.
3. **창의성에 집중할 수 있습니다.** 기술적 부분은 AI에게 맡기고, 나는 아이디어에 집중합니다.
4. **실패 비용이 거의 제로입니다.** 시간도 돈도 적게 들어서 마음 편하게 도전할 수 있습니다.

다음 챕터 미리보기

Chapter 3에서는 **“당신도 할 수 있는 이유”**를 이야기합니다.

- 나이가 오히려 장점인 이유
- 살아온 경험이 최고의 사업 아이디어인 이유
- 작게 시작하는 용기

"나는 너무 늦었다"고 생각하시나요?

다음 챕터를 읽고 나면 생각이 바뀔 겁니다.

Chapter 3. 당신도 할 수 있는 이유

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. **나이는 장벽이 아니라 장점**이라는 것을 이해합니다.
 2. 살아온 경험이 곧 **사업 아이디어**가 된다는 것을 알게 됩니다.
 3. **작게 시작하는 용기**가 가장 중요하다는 확신을 갖습니다.
-

1. 나이는 숫자에 불과합니다

너무 늙은 거 아닌가요?

“나이 50이 넘었는데 이제 와서 뭘 시작해?”

이런 생각이 드셨을 수 있습니다.

주변에서도 그렇게 말할 수 있습니다.

하지만 역사를 보면 다릅니다.

50대 이후에 시작한 사람들

레이 크록 — 맥도날드 (52세)

맥도날드를 만든 사람을 아시나요?

레이 크록이라는 사람입니다.

그는 52세에 맥도날드 사업을 시작했습니다.

그전까지 밀크셰이크 기계를 팔던 세일즈맨이었습니다.

특별한 배경이 없었습니다.

할랜드 샌더스 — KFC (65세)

KFC의 할아버지 로고를 아시죠?

실제 인물입니다.

할랜드 샌더스는 65세에 KFC를 시작했습니다.

그전에는 수십 가지 직업을 전전했습니다.

65세에 “이제 정말 해보자” 결심했습니다.

디지털 시대에는 더 쉽습니다

레이 크록과 할랜드 샌더스는 직접 가게를 열었습니다.

돈도 많이 들고, 위험도 컸습니다.

디지털 사업은 다릅니다.

- 가게를 열 필요가 없습니다
- 직원을 고용할 필요가 없습니다
- 재고를 쌓을 필요가 없습니다
- 컴퓨터 한 대면 시작할 수 있습니다

나이가 많다고 못 할 이유가 없습니다.

오히려 유리한 점이 더 많습니다.

2. 경험이 곧 아이디어

젊은 사람에게 없는 것

20대 개발자는 기술이 뛰어납니다.

코드를 빠르게 작성합니다.

새로운 도구를 금방 배웁니다.

하지만 한 가지가 부족합니다.

인생 경험입니다.

30년 넘게 직장 생활을 하셨나요?

자녀를 키워 보셨나요?

건강 문제를 겪어 보셨나요?

취미 활동을 오래 하셨나요?

이 모든 경험이 사업 아이디어입니다.

경험에서 아이디어 찾기

몇 가지 예를 들어 보겠습니다.

30년 회사원 경력이 있다면:

→ 직장인을 위한 업무 효율 도구

자녀 교육에 관심이 많다면:

→ 학부모를 위한 교육 정보 사이트

등산이 취미라면:

→ 등산 코스 추천 서비스

요리를 좋아한다면:

→ 레시피 공유 커뮤니티

건강 관리에 신경 쓴다면:

→ 건강 기록 앱

여러분만의 전문 분야가 있습니다

20대는 모든 분야의 초보입니다.

하지만 50대, 60대는 **최소 한 분야의 전문가**입니다.

그 전문 지식을 서비스로 만들면 됩니다.

기술은 AI가 담당합니다.

아이디어와 경험은 여러분이 담당합니다.

이것이 **최강의 조합입니다.**

3. 작게 시작하는 용기

처음부터 큰 걸 만들 필요 없습니다

"사업"이라고 하면 거창하게 느껴집니다.

사무실도 있어야 할 것 같고요.

직원도 뽑아야 할 것 같습니다.

하지만 그럴 필요 없습니다.

방 한쪽 구석, 노트북 한 대.

이것으로 충분합니다.

첫 번째 목표는 작게

첫 번째 목표를 이렇게 잡아보세요.

“간단한 웹페이지 하나 만들어보기”

이게 전부입니다.

수백만 원을 벌겠다는 목표가 아닙니다.

“뭔가 하나 만들어보기”가 첫 목표입니다.

하나를 만들면 두 번째가 쉬워집니다.

두 번째를 만들면 세 번째는 더 쉬워집니다.

완벽하지 않아도 됩니다

“좀 더 준비되면 시작할게.”

“조금만 더 공부하고.”

“좀 더 좋은 아이디어가 떠오르면.”

이런 말을 하며 시작을 미루지 마세요.

완벽한 준비란 없습니다.

80%만 준비되었으면 시작하세요.
나머지 20%는 하면서 채우면 됩니다.

실생활 비유: 씨앗 심기

큰 농장이 아니라 작은 화분에서

사업을 시작하는 것을 씨앗 심기에 비유해 보겠습니다.

처음부터 큰 농장을 살 필요 없습니다.
작은 화분 하나면 됩니다.

베란다에 화분을 놓습니다.
씨앗 하나를 심습니다.
물을 줍니다.
기다립니다.

싹이 트면 자신감이 생깁니다

며칠 후 작은 싹이 나옵니다.
"내가 키운 거다!"라는 기쁨이 있습니다.

그러면 화분을 하나 더 놓습니다.
다른 씨앗도 심어봅니다.

어떤 것은 잘 자라고 어떤 것은 시듭니다.
시든 것은 뽑고 새 씨앗을 심으면 됩니다.

화분이 늘어나면 작은 정원이 됩니다

하나에서 시작한 화분이 다섯 개가 됩니다.
다섯 개가 열 개가 됩니다.
어느새 베란다가 작은 정원이 됩니다.

사업도 마찬가지입니다.

작은 웹페이지 하나에서 시작합니다.

사용자가 모이면 기능을 추가합니다.

작은 프로젝트가 하나의 사업이 됩니다.

모든 큰 나무는 작은 씨앗에서 시작했습니다.

실제 사례: 늦게 시작해도 성공할 수 있습니다

기술이 아니라 경험이 무기입니다

Pieter Levels와 Marc Lou의 이야기를 기억하시나요?

그들은 젊은 나이에 시작했습니다.

하지만 그들에게는 AI가 없었습니다.

여러분에게는 두 가지 무기가 있습니다.

1. 수십 년의 인생 경험 — 아이디어의 원천

2. AI 도구 — 기술의 보조

Pieter Levels가 했던 것을 떠올려 보세요.

“사람들에게 필요한 것을 만들자.”

여러분은 사람들에게 뭐가 필요한지 이미 알고 있습니다.

수십 년간 살면서 느낀 불편함, 필요한 것들.

그것이 바로 사업 아이디어입니다.

AI가 기술 격차를 메워줍니다

“기술을 모르는데 어떡해?”

괜찮습니다.

AI가 메워줍니다.

코딩을 모르면 AI가 만들어 줍니다.
디자인을 모르면 AI가 도와줍니다.
마케팅 문구를 못 쓰면 AI가 써줍니다.

여러분은 방향만 정하면 됩니다.
실행은 AI와 함께 하면 됩니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 살아온 경험이 최고의 사업 아이디어입니다. 젊은 사람에게는 없는 통찰이 여러분에게 있습니다.
 2. 완벽하지 않아도 시작할 수 있습니다. 80%만 준비되면 나머지는 하면서 배웁니다.
 3. 작은 것부터 만들어보세요. 화분 하나에서 시작해 정원을 만들 수 있습니다.
 4. AI가 기술 격차를 메워줍니다. 코딩, 디자인, 마케팅 모두 AI가 도와줍니다.
-

다음 챕터 미리보기

Part 1이 끝나면 Part 1 종합 실습이 있습니다.
지금까지 배운 내용을 정리하고, 실제로 해볼 시간입니다.

그리고 Part 2에서는 드디어 실전입니다!

- Chapter 4: AI 도구를 설치하고 첫 대화를 나눕니다
- Chapter 5: 10분 만에 첫 웹페이지를 만듭니다
- Chapter 6: 만든 것을 인터넷에 공개합니다

이론은 끝났습니다.
이제 직접 해볼 차례입니다!

Part 1 마무리: 바이브 코딩 마인드셋 점검

Part 1 핵심 요약

세 개의 챕터를 통해 이런 것들을 배웠습니다.

Chapter 1 — AI 시대, 코딩의 새로운 패러다임

- 바이브 코딩은 코드를 직접 쓰는 것이 아닙니다.
- AI에게 내가 원하는 것을 **말로 설명**하는 것입니다.
- 코딩 지식보다 **아이디어**가 더 중요합니다.

Chapter 2 — 바이브 코딩 도구 살펴보기

- Claude, Cursor, Bolt 같은 AI 도구를 알게 되었습니다.
- 대부분 **무료**로 시작할 수 있습니다.
- 도구는 많지만, 하나만 잘 써도 충분합니다.

Chapter 3 — 바이브 코딩 마인드셋

- 완벽하지 않아도 **일단 시작**하는 것이 중요합니다.
- 작은 것부터 만들어 보는 것이 최고의 공부법입니다.
- 실패해도 괜찮습니다. AI가 다시 도와줍니다.

실습 과제: 내가 만들고 싶은 것 3가지 적어보기

지금 바로 해볼 수 있는 실습입니다.

어렵게 생각하지 마세요.

이렇게 해보세요

아래 빈칸에 내가 만들고 싶은 것을 적어 보세요.

웹사이트, 앱, 도구, 서비스 아무거나 괜찮습니다.

1. _____

이유: _____

2. _____

이유: _____

3. _____

이유: _____

잘 안 떠오르시나요? 힌트를 드릴게요

이런 것들을 생각해 보세요.

- **불편한 것:** “이런 게 있으면 편할 텐데…”
- **취미:** “내 취미를 다른 사람과 나누고 싶다”
- **경험:** “내가 아는 것을 정리해서 알려주고 싶다”
- **주변 사람:** “우리 동네에 이런 서비스가 있으면 좋겠다”

예시

만들고 싶은 것	이유
우리 동네 맛집 지도	블로그 리뷰는 광고가 많아서 진짜 맛집만 모으고 싶다
하루 운동 기록 사이트	종이 다이어리보다 편하게 기록하고 싶다
증고 물품 나눔 게시판	우리 아파트 단지에서 쓸 수 있는 나눔 공간이 필요하다

정답은 없습니다.

떠오르는 대로 적어 보세요.

이 목록이 앞으로 여러분의 첫 프로젝트가 됩니다.

자가 진단 체크리스트

Part 1을 잘 이해했는지 스스로 점검해 보세요.

아래 항목을 읽고 체크해 보세요.

- [] 바이브 코딩이 뭔지 한 문장으로 설명할 수 있다.
→ “AI에게 내가 원하는 것을 말로 설명해서 만드는 것”
- [] AI 코딩 도구를 최소 2개 이상 말할 수 있다.
→ 예: Claude, Cursor, Bolt, Lovable 등
- [] 코딩을 몰라도 시작할 수 있다는 것을 이해했다.
→ 중요한 건 코드가 아니라 아이디어입니다
- [] 만들고 싶은 것이 최소 1개는 떠올랐다.
→ 위 실습 과제에 적어 보셨나요?
- [] "나도 할 수 있겠다"는 마음이 조금이라도 생겼다.
→ 이 마음이 가장 중요합니다!

결과 확인

- **5개 모두 체크**: 완벽합니다! Part 2로 바로 넘어가세요.
 - **3~4개 체크**: 잘하고 있습니다. 부족한 부분만 다시 읽어 보세요.
 - **1~2개 체크**: 괜찮습니다. Chapter 1부터 천천히 다시 읽어 보세요.
 - **0개 체크**: 걱정 마세요. 이 책은 도망가지 않습니다. 편하게 다시 시작하세요.
-

당신도 할 수 있습니다

혹시 이런 생각이 드시나요?

“나이가 많은데 할 수 있을까?”
“컴퓨터를 잘 못 하는데…”
“젊은 사람들이나 하는 거 아닌가?”

한 가지 이야기를 더 해드리겠습니다.

Pieter Levels는 특별한 사람이 아니었습니다

Chapter 1에서 소개한 Pieter Levels를 기억하시나요?

- 부모님 집에서 시작했습니다
- 우울증을 겪고 있었습니다
- 특별한 학위도 없었습니다

그런데 지금은 연 38억 원을 벌고 있습니다.

Marc Lou도 마찬가지입니다

Marc Lou(마크 루)라는 사람도 있습니다.

그는 여러 번 실패한 후에 성공했습니다.

그의 전략은 단순했습니다.

“작은 제품을 빠르게 만들고,
반응이 있는 것만 키운다.”

그는 이 방법으로 월 수천만 원을 벌고 있습니다.
역시 혼자서 운영합니다.

그들과 여러분의 차이

Pieter Levels와 Marc Lou가 시작할 때는 AI가 없었습니다.
직접 코딩을 배워야 했습니다.

여러분에게는 AI가 있습니다.

그들보다 유리한 조건에서 시작하는 것입니다.

나이는 상관없습니다.
경험은 오히려 장점입니다.
살아온 세월만큼 아이디어가 많다는 뜻이니까요.

Part 2 미리보기: 본격적으로 시작하기

Part 1에서는 마음가짐을 다렸습니다.

Part 2에서는 직접 손을 움직여 봅니다.

- Chapter 4: AI 도구 설치하고 첫 대화 나누기
- Chapter 5: 나의 첫 웹페이지 만들기 (10분 안에!)
- Chapter 6: 만든 것을 인터넷에 공개하기

Part 2를 마치면 **실제로 동작하는 웹페이지**를 갖게 됩니다.

인터넷에 공개해서 누구나 볼 수 있게 만듭니다.

준비되었나요?

다음 장을 넘겨 주세요.

Chapter 4. AI 도구 설치하고 첫 대화 나누기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. Claude에 가입하고 사용하는 방법을 알게 됩니다.
 2. Cursor를 설치하는 방법을 알게 됩니다.
 3. AI와 첫 대화를 나눌 수 있습니다.
-

1. Claude 가입하기

Claude가 뭔가요?

Claude(클로드)는 AI 대화 도구입니다.

사람처럼 대화할 수 있습니다.

질문하면 답해줍니다.

코드도 작성해 줍니다.

카카오톡으로 친구에게 부탁하는 것과 비슷합니다.

"이거 만들어줘"라고 하면 만들어 줍니다.

가입 방법 (5분이면 됩니다)

1단계: 사이트 접속

인터넷 브라우저를 엽니다.

주소창에 claude.ai를 입력합니다.

엔터를 누릅니다.

2단계: 회원가입

화면에 “Sign up” 버튼이 보입니다.

이메일 주소를 입력합니다.

비밀번호를 정합니다.

이메일로 온 인증 번호를 입력합니다.

구글 계정이 있다면 더 쉽습니다.

“Continue with Google”을 누르면 됩니다.

3단계: 완료!

가입이 끝나면 대화 화면이 나타납니다.

아래쪽에 입력창이 보입니다.

여기에 글을 쓰면 AI가 답합니다.

무료로 사용할 수 있습니다.

처음에는 무료 버전으로 충분합니다.

2. Cursor 설치하기

Cursor가 뭘까요?

Cursor(커서)는 코드 편집기입니다.

코드 편집기란 코드를 쓰는 프로그램입니다.

메모장의 전문가 버전이라고 생각하시면 됩니다.

Cursor의 특별한 점은 [AI가 내장](#)되어 있다는 것입니다.

편집기 안에서 바로 AI에게 말할 수 있습니다.

설치 방법

1단계: 다운로드

인터넷 브라우저에서 cursor.com에 접속합니다.

“Download” 버튼을 누릅니다.

내 컴퓨터(Windows/Mac)에 맞는 버전이 자동으로 다운됩니다.

2단계: 설치

다운받은 파일을 더블클릭합니다.

“다음”, “동의”, “설치” 버튼을 누릅니다.

보통 1~2분이면 설치가 끝납니다.

3단계: 실행

설치가 끝나면 Cursor를 실행합니다.

처음 화면이 조금 복잡해 보일 수 있습니다.

걱정하지 마세요.

우리가 쓸 기능은 몇 가지 안 됩니다.

지금 당장 안 해도 괜찮습니다

Cursor는 Part 3에서 본격적으로 사용합니다.

지금은 Claude만 있어도 충분합니다.

[Claude에 먼저 익숙해지는 것을 추천합니다.](#)

3. AI와 첫 대화 나누기

긴장하지 마세요

AI와 대화하는 것은 어렵지 않습니다.

카카오톡으로 메시지 보내는 것과 같습니다.

틀려도 괜찮습니다.

이상한 질문을 해도 괜찮습니다.

AI는 절대 화내지 않습니다.

첫 대화 해보기

Claude 화면 아래 입력창에 이렇게 써보세요.

“안녕, 나는 코딩을 처음 배우려는 사람이야.
쉽게 설명해줘.”

엔터를 누르면 Claude가 답합니다.

친절하게 설명해 줄 겁니다.

이런 것들도 물어보세요

Claude에게 다양한 것을 물어볼 수 있습니다.

일상적인 질문:

“오늘 저녁 메뉴 추천해줘.”

정보 찾기:

“고혈압에 좋은 음식이 뭐야?”

코딩 관련:

“웹사이트가 뭔지 쉽게 설명해줘.”

아이디어 상담:

“50대가 시작할 수 있는 온라인 사업 아이디어 알려줘.”

무엇이든 물어보세요.

연습이 가장 좋은 공부입니다.

대화할 때 팁

팁 1: 한국어로 말하세요

Claude는 한국어를 잘 이해합니다.

영어로 말할 필요 없습니다.

팁 2: 구체적으로 말하세요

“알려줘” (X)

“50대 초보자에게 쉽게 알려줘” (O)

팁 3: 모르면 다시 물어보세요

답이 어려우면 이렇게 말하세요.

“더 쉽게 설명해줘.”

“예를 들어서 설명해줘.”

실생활 비유: 새 스마트폰 처음 켜기

처음엔 다 낯설었습니다

스마트폰을 처음 쓰셨을 때를 떠올려 보세요.

화면을 어떻게 터치하는지도 몰랐습니다.

앱이 뭔지도 몰랐습니다.

하지만 전화 거는 법부터 배웠습니다.

문자 보내는 법을 배웠습니다.

카카오톡을 깔았습니다.

한 달 뒤에는 능숙하게 쓰고 있었습니다.

AI 도구도 마찬가지입니다

처음엔 낯설어 보입니다.

뭘 어떻게 해야 할지 모릅니다.

하지만 대화 한 번, 두 번 하다 보면 익숙해집니다.

일주일이면 편하게 쓸 수 있습니다.

스마트폰을 배웠던 것처럼 AI 도구도 배울 수 있습니다.

처음이 가장 어렵습니다. 시작만 하세요.

실제 사례: 처음 AI를 써본 사람들의 반응

“이게 정말 되는 거야?”

AI를 처음 써본 분들의 반응은 대부분 비슷합니다.

“세상에, 이게 진짜 돼?”

“사람이 대답하는 줄 알았어!”

“이걸 왜 이제서야 알았지?”

60대 어머니에게 Claude를 보여드린 아들의 이야기입니다.

“어머니가 ‘김치찌개 맛있게 끓이는 법’ 물어보셨어요.

Claude가 상세하게 알려주니까

‘이 친구 요리 잘 아네?’ 하셨어요.”

그 뒤로 어머니는 매일 Claude에게 물어보신다고 합니다.

건강 정보도 물어보고, 여행지도 추천받으신다고 합니다.

AI는 친구같은 도우미입니다

AI를 어렵게 생각하지 마세요.

모든 것을 알고 있는 친절한 친구라고 생각하세요.

이 친구는 24시간 언제든 대답합니다.

화를 내지 않습니다.

같은 질문을 해도 싫어하지 않습니다.

한국어로 말하면 한국어로 답합니다.

이보다 좋은 도우미가 있을까요?

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. Claude는 무료로 시작할 수 있습니다. claude.ai에서 5분이면 가입됩니다.
 2. Cursor는 AI가 내장된 코드 편집기입니다. 나중에 설치해도 괜찮습니다.
 3. AI에게 한국어로 말하면 됩니다. 어렵게 생각하지 마세요.
 4. 처음엔 간단한 질문부터 시작하세요. "안녕"이라고 말하는 것도 훌륭한 첫걸음입니다.
-

다음 챕터 미리보기

Chapter 5에서는 드디어 첫 웹페이지를 만들어 봅니다!

- AI에게 웹페이지를 요청하는 방법
- 10분 만에 완성하는 비결
- 마음에 안 드는 부분을 수정하는 방법

정말 10분이면 될까요?

다음 챕터에서 직접 경험해 보세요!

Chapter 5. 10분 만에 첫 웹페이지 만들기

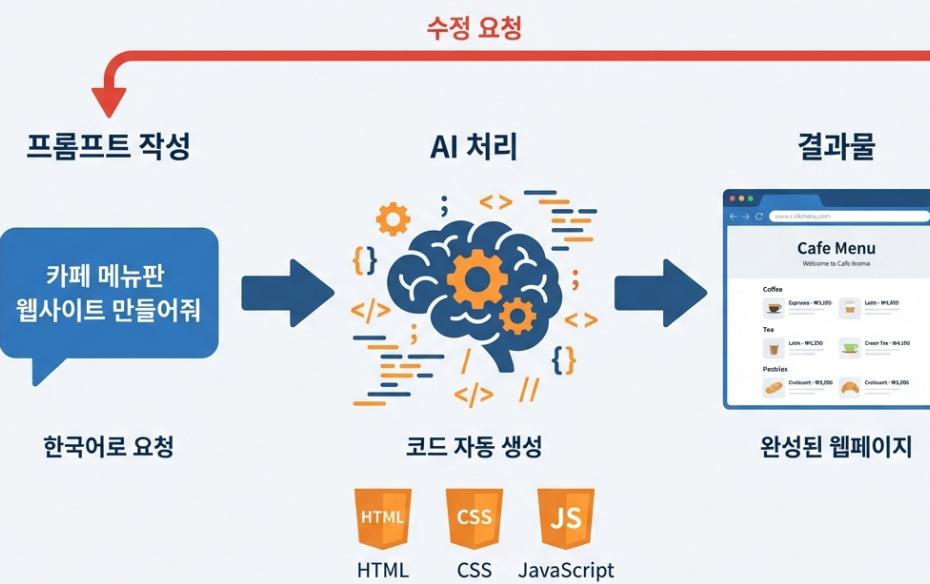
학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. AI에게 웹페이지를 요청하는 방법을 알게 됩니다.
2. 결과물을 확인하고 수정 요청하는 방법을 배웁니다.
3. 실제로 10분 안에 나만의 웹페이지를 만들 수 있습니다.

1. AI에게 웹페이지 요청하기

프롬프트에서 웹페이지까지



프롬프트가 곧 설계도입니다

AI에게 하는 요청을 **프롬프트**라고 합니다.

건축에 비유하면 설계도와 같습니다.

설계도가 구체적일수록 좋은 집이 나옵니다.

프롬프트가 구체적일수록 좋은 웹페이지가 나옵니다.

첫 번째 프롬프트 따라하기

Claude에게 이렇게 말해 보세요.

“나의 자기소개 웹페이지를 만들어줘.
이름은 홍길동이고, 취미는 등산이야.
좋아하는 말은 '천리길도 한 걸음부터'야.
예쁜 디자인으로 만들어줘.”

이름과 취미는 여러분의 것으로 바꾸세요.

엔터를 누르면 Claude가 HTML 코드를 만들어 줍니다.

HTML이란 웹페이지를 만드는 언어입니다.

이 코드를 이해할 필요는 없습니다.

결과물 확인하기

Claude가 만든 코드를 복사합니다.

메모장을 열고 붙여넣습니다.

파일 이름을 **index.html**로 저장합니다.

저장한 파일을 더블클릭하면?

웹페이지가 열립니다!

내 이름이 보입니다.

취미가 보입니다.

좋아하는 말이 보입니다.

이것이 여러분의 첫 웹페이지입니다.

2. 수정 요청하기

마음에 안 드는 부분이 있다면

첫 결과가 완벽하지 않을 수 있습니다.

색상이 마음에 안 들 수 있습니다.

글씨가 너무 작을 수 있습니다.

AI에게 다시 말하면 됩니다.

수정 프롬프트 예시

이렇게 요청해 보세요.

“배경색을 하늘색으로 바꿔줘.”

“글씨 크기를 더 크게 해줘.”

“내 사진을 넣을 자리를 추가해줘.”

“아래쪽에 연락처 정보도 넣어줘.”

한 번에 하나씩 요청하는 게 좋습니다.

AI가 수정된 코드를 다시 만들어 줍니다.

몇 번이든 수정할 수 있습니다

수정에 제한이 없습니다.

10번을 수정해도 괜찮습니다.

20번을 수정해도 괜찮습니다.

AI는 지치지 않습니다.

마음에 들 때까지 요청하세요.

3. 더 멋진 웹페이지 만들기

좀 더 구체적인 프롬프트

기본 자기소개 페이지에 만족하셨나요?

조금 더 욕심을 내볼까요?

이렇게 요청해 보세요.

"나의 자기소개 웹페이지를 만들어줘.

포함할 내용:

- 이름: [이름]
- 한 줄 소개: [소개]
- 취미 3가지: [취미1], [취미2], [취미3]
- 좋아하는 명언: [명언]
- 연락처: [이메일]

디자인:

- 색상은 따뜻한 느낌으로
- 글씨는 크고 읽기 쉽게
- 사진 넣을 공간 포함

- 모바일에서도 잘 보이게"

구체적으로 요청할수록 결과가 좋아집니다.

다른 종류의 페이지도 만들 수 있습니다

자기소개 말고도 다양한 페이지를 만들 수 있습니다.

"우리 동네 맛집 3곳을 소개하는 페이지 만들어줘."

"내 강아지 사진첩 웹페이지 만들어줘."

"우리 동호회 소개 페이지 만들어줘."

아이디어가 떠오르면 AI에게 말해보세요.

10분이면 무엇이든 만들어 줍니다.

실생활 비유: 레고 블록으로 집 만들기

설명서만 있으면 누구나 조립합니다

레고를 아시나요?

작은 블록을 조립해서 집을 만드는 장난감입니다.

레고에는 [설명서](#)가 들어있습니다.

설명서대로 따라하면 누구나 멋진 집을 만들 수 있습니다.

건축 지식이 필요 없습니다.

설명서만 따라하면 됩니다.

프롬프트가 설명서입니다

바이브 코딩에서 프롬프트는 레고 설명서와 같습니다.

“파란색 지붕 집을 만들어줘.”

이렇게 말하면 AI가 블록을 조립해 줍니다.

“지붕 색을 빨간색으로 바꿔줘.”

이렇게 말하면 AI가 블록을 교체해 줍니다.

마음에 안 들면 부분만 바꾸면 됩니다.

처음부터 다시 만들 필요 없습니다.

실제 사례: 바이브 코딩 10분 챌린지

실제로 10분이면 됩니다

“10분이라니, 과장 아니야?”

실제로 바이브 코딩으로 웹페이지를 만드는 과정입니다.

- **0~2분**: Claude에게 프롬프트 작성
- **2~4분**: AI가 코드 생성 + 확인
- **4~7분**: 수정 요청 1~2번
- **7~10분**: 최종 확인 + 저장

정말 10분이면 완성됩니다.

전통적 코딩이었다면?

HTML, CSS를 배우는 데만 2주가 걸립니다.

같은 페이지를 직접 만드는 데 하루는 걸립니다.

10분 vs 2주.

이것이 바이브 코딩의 힘입니다.

처음이라 15분 걸려도 괜찮습니다

물론 처음이라 익숙하지 않을 수 있습니다.

15분이 걸릴 수도 있고 20분이 걸릴 수도 있습니다.

전혀 괜찮습니다.

두 번째는 10분이 됩니다.

세 번째는 5분이면 됩니다.

점점 빨라집니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 프롬프트가 곧 설계도입니다. 구체적으로 쓸수록 좋은 결과가 나옵니다.
 2. 처음부터 완벽할 필요 없습니다. 기본을 만들고 수정하면 됩니다.
 3. 수정은 AI에게 다시 요청하면 됩니다. 몇 번이든 괜찮습니다.
 4. 10분이면 기본 웹페이지가 완성됩니다. 직접 해보면 믿게 됩니다.
-

다음 챕터 미리보기

Chapter 6에서는 만든 웹페이지를 인터넷에 공개합니다!

- Netlify로 무료 배포하기
- 나만의 인터넷 주소 갖기
- 가족과 친구에게 링크 공유하기

만든 것을 세상에 보여줄 시간입니다.

정말 뿌듯한 경험이 될 거예요!

Chapter 06. 내 웹페이지를 세상에 공개하기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. Netlify를 사용해 웹페이지를 인터넷에 올릴 수 있습니다.
 2. 도메인(인터넷 주소)의 개념을 이해합니다.
 3. 배포한 웹페이지 링크를 다른 사람과 공유할 수 있습니다.
-

1. 배포란 무엇인가요?

내 컴퓨터 안에 갇힌 웹페이지

지금까지 열심히 웹페이지를 만드셨죠?

그런데 이 페이지는 내 컴퓨터에만 있어요.

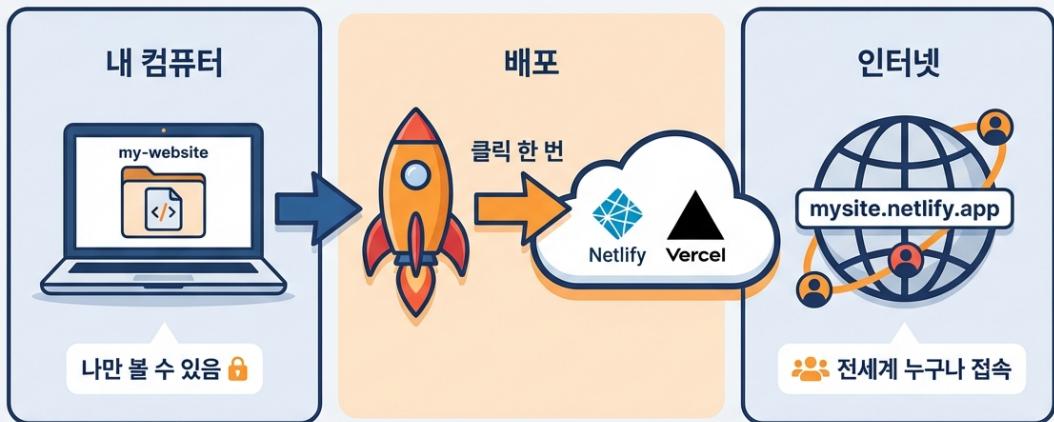
옆 사람도 볼 수 없는 상태입니다.

배포란 내 웹페이지를 인터넷에 올리는 거예요.

영어로는 Deploy(디플로이)라고 해요.

올리고 나면 전 세계 누구나 볼 수 있어요.

배포: 내 컴퓨터에서 인터넷으로



왜 배포를 해야 할까요?

만든 것을 보여줘야 의미가 생겨요.

혼자만 보면 발전이 느립니다.

다른 사람의 반응이 성장의 시작이에요.

"이 부분이 좋아요!"라는 말을 들으면요.

자신감이 생기고 의욕이 올라갑니다.

빨리 공개하는 것이 중요한 이유예요.

2. Netlify로 배포하기

Netlify가 뭔가요?

Netlify(넷틀리파이)는 무료 배포 서비스예요.

웹페이지를 인터넷에 올려주는 곳이에요.

복잡한 설정 없이 바로 사용할 수 있어요.

[드래그 앤 드롭](#)으로 배포가 됩니다.

마우스로 파일을 끌어다 놓는 거예요.

정말 클릭 몇 번이면 끝나요.

1단계: Netlify 가입하기

인터넷 브라우저를 열어주세요.

주소창에 netlify.com을 입력합니다.

화면에 “Sign up” 버튼이 보일 거예요.

이메일 주소로 가입할 수 있어요.

구글 계정으로도 가입이 가능합니다.

가입은 1분이면 충분해요.

2단계: 대시보드 확인하기

가입하면 대시보드 화면이 나타나요.

[대시보드](#)란 관리 화면을 말해요.

내 사이트를 관리하는 곳이에요.

화면에 “Sites” 메뉴가 보일 겁니다.

여기서 새 사이트를 올릴 수 있어요.

아래쪽에 점선 박스가 보여요.

3단계: 폴더 드래그하기

내 컴퓨터에서 웹페이지 폴더를 찾으세요.

보통 `dist` 폴더에 파일이 있어요.

`dist`는 배포용 파일이 담긴 폴더예요.

`dist` 폴더가 없다면 걱정 마세요.

HTML 파일이 있는 폴더면 됩니다.

그 폴더를 점선 박스에 끌어다 놓으세요.

4단계: 배포 완료!

폴더를 놓으면 업로드가 시작돼요.

보통 30초에서 1분이면 끝나요.

초록색 “Published” 표시가 나타납니다.

축하드려요! 배포가 완료되었어요.

이제 전 세계 누구나 볼 수 있습니다.

정말 간단하지 않나요?

3. 도메인 이야기

도메인이란?

도메인은 인터넷 주소를 말해요.

`naver.com` 네이버의 도메인이에요.

`google.com` 구글의 도메인이고요.

집에도 주소가 있잖아요.

웹페이지에도 주소가 있어야 해요.

그 주소가 바로 도메인입니다.

무료 주소로 시작하세요

Netlify에서 배포하면 주소가 자동으로 생겨요.

예를 들어 [my-site.netlify.app](#) 같은 거예요.

이것이 내 웹페이지의 무료 주소입니다.

나중에 원하면 직접 도메인을 살 수 있어요.

[my-shop.com](#) 같은 주소를 가질 수 있죠.

하지만 처음엔 무료 주소로 충분해요.

링크 공유하기

배포가 끝나면 주소를 복사하세요.

카카오톡에 붙여넣어 보내보세요.

"내가 만든 웹페이지야!"라고요.

상대방이 링크를 누르면 바로 보여요.

스마트폰에서도 잘 열립니다.

그 순간이 정말 뿌듯할 거예요.

실생활 비유: 가게 문 열기

식당을 준비하는 상황을 떠올려 보세요.

인테리어를 완벽하게 하고 싶으시겠죠.

메뉴판도 예쁘게 만들고 싶을 거예요.

그런데 문을 안 열면 어떻게 될까요?

손님이 올 수가 없어요.

어떤 음식이 인기인지 알 수도 없어요.

문을 여는 것이 가장 중요합니다.

인테리어가 완벽하지 않아도 괜찮아요.

손님이 와야 뭘 고칠지 알 수 있거든요.

웹페이지도 똑같아요.

완벽하지 않아도 일단 공개하세요.

사람들이 봐야 뭘 개선할지 알게 됩니다.

실제 사례: Pieter Levels의 첫 배포 이야기

Pieter Levels(피터 레벨스)를 기억하시죠?

NomadList라는 서비스를 만든 사람이에요.

디지털 노마드를 위한 도시 정보 사이트예요.

디지털 노마드란 어떤 사람일까요?

노트북 하나로 여행하며 일하는 사람이에요.

NomadList는 살기 좋은 도시를 알려줘요.

그런데 처음 배포했을 때 어땠을까요?

완벽하지 않았어요.

디자인도 투박하고 기능도 부족했어요.

단순한 스프레드시트 수준이었어요.

도시 이름과 몇 가지 정보만 있었죠.

지금 보면 초라할 정도였어요.

하지만 그는 일단 인터넷에 올렸어요.

사람들에게 링크를 공유했어요.

"이런 걸 만들었는데 봐줘"라고요.

그랬더니 사람들이 반응하기 시작했어요.

"이 기능이 있으면 좋겠어요."

"이 도시 정보도 추가해주세요."

피드백을 받으며 서비스가 점점 좋아졌어요.

지금은 수십억 원 매출을 올리고 있어요.

그 시작은 불완전한 첫 배포였습니다.

피터 레벨스의 유명한 말이 있어요.

“만든 것을 세상에 보여줘야 피드백을 받을 수 있다.”

완벽할 때까지 기다리면요.

영원히 시작 못 해요.

일단 올리는 것이 답이에요.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 배포는 클릭 몇 번이면 끝납니다. Netlify에 폴더를 끌어다 놓으면 자동으로 올라가요.
2. 무료 도메인으로 시작할 수 있어요. Netlify가 무료 주소를 제공하니 비용 걱정 없습니다.
3. 전 세계 누구나 볼 수 있어요. 링크만 공유하면 어디서든 접속할 수 있습니다.
4. 완성 후 공유가 자신감을 줍니다. 내가 만든 것을 보여주는 경험은 큰 동기부여가 돼요.

다음 챕터 미리보기

축하드려요! Part 2가 거의 끝났어요.

다음은 Part 2 종합 실습입니다.

지금까지 배운 것을 모두 활용해 봅니다.

직접 웹페이지를 처음부터 만들고요.

Netlify에 배포까지 해보는 시간이에요.

한 단계씩 차근차근 함께 해봐요!

Part 2 마무리: 첫 웹사이트 만들기 프로젝트

Part 2 핵심 요약

세 개의 챕터를 통해 이런 것들을 배웠습니다.

Chapter 4 — AI 도구 설치하고 첫 대화 나누기

- Claude, Cursor 등 AI 도구에 [가입하는 방법](#)을 배웠습니다.
- 처음 가입할 때는 [무료 플랜](#)으로 충분합니다.
- AI에게 말을 거는 것은 친구에게 부탁하는 것과 비슷합니다.

Chapter 5 — 나의 첫 웹페이지 만들기

- AI에게 부탁하면 [10분 안에](#) 웹페이지를 만들 수 있습니다.
- HTML, CSS 같은 용어를 몰라도 괜찮습니다.
- 내가 원하는 대로 [수정도](#) [AI에게 부탁](#)하면 됩니다.

Chapter 6 — 만든 것을 인터넷에 공개하기

- Netlify를 사용하면 [무료로 배포](#)할 수 있습니다.
 - 배포하면 [나만의 인터넷 주소](#)가 생깁니다.
 - 그 주소를 누구에게든 공유할 수 있습니다.
-

실습 과제: 나만의 자기소개 웹페이지를 만들고 배포하기

지금 바로 해볼 수 있는 실습입니다.

천천히 따라 하시면 됩니다.

Step 1: Claude에게 자기소개 페이지 요청하기

Claude에게 이렇게 말해 보세요.

“나를 소개하는 예쁜 웹페이지를 만들어 줘.
이름은 _____이고, 취미는 _____이야.
따뜻한 느낌으로 만들어 줘.”

아래에 여러분의 정보를 먼저 적어 보세요.

항목	내 정보
이름	_____
한 줄 소개	_____
취미	_____
좋아하는 것	_____
하고 싶은 말	_____

Step 2: 내용 수정하기

AI가 만들어 준 페이지를 보고 수정할 부분을 찾아보세요.

이렇게 부탁하면 됩니다.

“배경색을 하늘색으로 바꿔 줘.”

“글씨를 좀 더 크게 해줘.”

“사진 넣는 자리를 추가해 줘.”

수정은 몇 번이든 괜찮습니다.

AI는 지치지 않습니다.

Step 3: Netlify에 배포하기

1. [netlify.com](https://www.netlify.com)에 접속합니다.
2. 회원가입을 합니다. (무료입니다.)
3. AI가 만들어 준 파일을 끌어다 놓습니다.
4. 몇 초 기다리면 [나만의 주소](#)가 생깁니다.

내 웹페이지 주소: [https://_____.netlify.app](https://______.netlify.app)

Step 4: 가족이나 친구에게 공유하기

이것이 가장 중요한 단계입니다.

카카오톡이나 문자로 링크를 보내 보세요.

“내가 만든 거야!” 한마디면 됩니다.

공유한 사람: _____

그 사람의 반응: _____

반응이 어떻든 괜찮습니다.

중요한 것은 [내가 만들고, 공유했다](#)는 사실입니다.

자가 진단 체크리스트

Part 2를 잘 마쳤는지 스스로 점검해 보세요.

아래 항목을 읽고 체크해 보세요.

- [] AI 도구(Claude 등)에 가입했다.
→ 가입만 해도 큰 첫 걸음입니다
- [] AI와 첫 대화를 나눠 봤다.
→ 무엇이든 좋습니다. "안녕"이라도 괜찮습니다
- [] 웹페이지를 하나 만들어 봤다.
→ 완벽하지 않아도 됩니다. 만들어 본 것 자체가 대단합니다
- [] 만든 웹페이지를 인터넷에 배포해 봤다.
→ 나만의 주소가 생겼다면 성공입니다
- [] 누군가에게 내 웹페이지를 공유했다.
→ 가족, 친구, 누구든 한 명이면 충분합니다

결과 확인

- 5개 모두 체크: 멋집니다! 이미 많은 사람보다 앞서 있습니다.
- 3~4개 체크: 아주 잘하고 있습니다. 나머지는 천천히 해보세요.
- 1~2개 체크: 괜찮습니다. AI 도구 가입부터 다시 시작해 보세요.
- 0개 체크: 걱정 마세요. Chapter 4를 다시 읽으면서 따라 해보세요.

당신도 할 수 있습니다

혹시 이런 생각이 드시나요?

“겨우 웹페이지 하나 만든 건데…”

“이걸로 뭘 할 수 있을까?”

한 가지 이야기를 해드리겠습니다.

Marc Lou의 첫 제품은 아주 간단했습니다

Marc Lou(마크 루)를 기억하시나요?

월 수천만 원을 벼는 1인 개발자입니다.

그의 첫 제품이 뭐였는지 아시나요?

아주 단순한 웹페이지 하나였습니다.

거창한 앱이 아니었습니다.

화려한 기능도 없었습니다.

“작게 시작하라.

완벽할 필요 없다.

일단 세상에 내놓아라.”

이것이 Marc Lou의 철학입니다.

여러분도 이미 시작했습니다

웹페이지를 만들고 배포했다면,

여러분은 이미 Marc Lou의 첫 걸음과 같은 일을 한 것입니다.

시작이 반입니다.

아니, 시작이 전부일 수도 있습니다.

작은 웹페이지 하나가 큰 변화의 씨앗이 됩니다.

Part 3 미리보기: 본격 프로젝트 시작하기

Part 2에서는 첫 웹페이지를 만들어 봤습니다.

Part 3에서는 진짜 나만의 프로젝트를 만들어 봅니다.

- Chapter 7: 아이디어를 구체적인 기획으로 바꾸기
- Chapter 8: AI에게 기능을 하나씩 추가해 달라고 요청하기
- Chapter 9: 보기 좋게 디자인 다듬기
- Chapter 10: 다른 사람에게 보여주고 피드백 받기

Part 3를 마치면 [실제 사용자가 쓸 수 있는 서비스](#)가 만들어집니다.

아이디어가 현실이 되는 과정을 함께 경험해 봅시다.

준비되셨나요?

다음 장을 넘겨 주세요.

Chapter 07. 실전 프로젝트 기획하기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. 내 일상의 불편함에서 사업 아이디어를 찾을 수 있습니다.
 2. MVP(최소 기능 제품)의 개념을 이해하고 적용할 수 있습니다.
 3. 핵심 기능 3가지를 정해서 프로젝트를 기획할 수 있습니다.
-

1. 아이디어는 불편함에서 시작합니다

거창할 필요 없어요

“어떤 서비스를 만들어야 할까요?”

이 질문에 막막하신 분이 많을 거예요.

대단한 아이디어가 필요한 건 아니에요.

가장 좋은 아이디어의 출발점은요.

바로 **내가 느끼는 불편함**이에요.

"이런 게 있으면 편할 텐데"가 시작이에요.

불편함을 찾는 방법

하루 동안 이런 순간을 떠올려 보세요.

“아, 이거 불편하다” 싶은 순간이요.

그게 바로 사업 아이디어입니다.

예를 들어 볼까요?

- “동네 맛집을 한눈에 보고 싶다”

- “우리 동호회 일정 관리가 어렵다”
- “매일 먹는 약을 자꾸 잊어버린다”

이런 작은 불편함이 서비스가 돼요.
나만 불편한 게 아닐 수 있거든요.
같은 불편함을 느끼는 사람이 고객이에요.

종이에 적어보세요

지금 바로 종이를 꺼내보세요.
불편한 것 5개만 적어보세요.
쓰다 보면 아이디어가 구체화돼요.

완벽한 아이디어는 없어요.
일단 적는 것이 시작입니다.
나중에 골라도 충분해요.

2. MVP: 핵심 기능만 담은 첫 버전

MVP — 최소 기능 제품



MVP가 뭔가요?

MVP는 Minimum Viable Product의 줄임말이에요.

한국어로는 "최소 기능 제품"이라고 해요.

꼭 필요한 기능만 담은 첫 버전이에요.

쉽게 말해서 이런 거예요.

100점짜리를 만들려 하지 마세요.

60점짜리를 빨리 만드는 것이 MVP예요.

왜 MVP가 중요한가요?

완벽하게 만들려면 시간이 오래 걸려요.

몇 달씩 만들다가 지쳐서 포기해요.

그런데 완성해도 사람들이 안 쓸 수 있어요.

MVP는 빠르게 만들어서 빨리 보여주는 거예요.
사람들이 원하는지 먼저 확인하는 거죠.
확인한 다음에 더 만들어도 늦지 않아요.

핵심 기능 3가지만 정하세요

아이디어가 정해졌다면요.
가장 중요한 기능 3가지만 고르세요.
3가지면 충분합니다.

예를 들어 "동네 맛집 사이트"라면요.

1. 맛집 목록 보여주기
2. 지도에서 위치 확인하기
3. 별점 남기기

이 3가지면 서비스가 됩니다.
예약 기능, 사진 업로드, 리뷰 기능은요.
나중에 추가해도 돼요.

실생활 비유: 식당 메뉴판

처음 여는 식당을 상상해 보세요.
메뉴를 몇 가지로 시작할까요?
3가지면 충분합니다.

김치찌개, 된장찌개, 비빔밥.
이 3가지로 문을 여세요.
손님이 오면 반응을 살피는 거예요.

“김치찌개가 제일 인기구나.”
그러면 김치찌개를 더 맛있게 만들어요.
“불고기도 있으면 좋겠어요”라는 요청이 오면요.
그때 메뉴를 추가하면 됩니다.

처음부터 100가지 메뉴를 준비하면 어떨까요?

재료비가 엄청 들어요.

음식 품질도 떨어져요.

이것이 바로 실패 레시피예요.

웹 서비스도 똑같습니다.

기능 3가지로 시작하세요.

잘 되는 것을 파악한 후에 확장하세요.

실제 사례: Marc Lou의 ShipFast

Marc Lou(마크 루)라는 사람이 있어요.

ShipFast라는 서비스를 만들었어요.

개발자들이 빠르게 제품을 만드는 템플릿이에요.

템플릿이란 미리 만들어진 틀을 말해요.

요리로 치면 반조리 식품 같은 거예요.

기본 틀이 있으니 완성이 빠른 거죠.

Marc Lou도 처음부터 대단하지 않았어요.

ShipFast의 첫 버전은 기능이 3개였어요.

1. 회원가입 기능

2. 결제 기능

3. 이메일 발송 기능

딱 이 3가지로 출시했어요.

“이 정도면 사람들이 쓸까?” 싶었대요.

그런데 반응이 왔어요.

사용자들이 요청하기 시작했어요.

“SEO 기능 추가해주세요.”

“블로그 기능도 있으면 좋겠어요.”

그 피드백을 받아서 하나씩 추가했어요.

지금은 월 수천만 원의 매출을 올려요.

시작은 기능 3개짜리 MVP였습니다.

여기서 배울 점이 뭘까요?

완벽한 기획보다 빠른 실행이 중요해요.

사용자 1명이라도 먼저 만족시키세요.

3. 내 프로젝트 기획하기

기획서 작성법

거창한 문서가 필요한 건 아니에요.

종이 한 장이면 충분합니다.

이 세 가지만 적으세요.

1. 한 줄 설명

“이 서비스는 __을 위한 __이다.”

예: “이 서비스는 동네 주민을 위한 맛집 지도이다.”

2. 핵심 기능 3가지

가장 중요한 기능만 고르세요.

나머지는 나중에 추가하면 돼요.

3. 첫 번째 사용자

누가 이 서비스를 가장 먼저 쓸까요?

가족? 친구? 동호회 회원?

딱 1명을 떠올려 보세요.

완벽하지 않아도 괜찮아요

기획이 완벽할 수 없어요.

만들면서 계속 바뀌는 게 정상이에요.

중요한 건 일단 시작하는 거예요.

기획에 일주일을 쓰지 마세요.

30분이면 충분합니다.

나머지는 만들면서 알게 돼요.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. **아이디어는 내 불편함에서 시작해요.** 거창한 아이디어보다 작은 불편함이 좋은 출발점입니다.
 2. **MVP는 핵심 기능만 담은 첫 버전이에요.** 60점짜리를 빨리 만들어서 빨리 보여주세요.
 3. **완벽한 기획보다 빠른 실행이 중요해요.** 기획은 30분이면 충분합니다.
 4. **사용자 1명이라도 먼저 만족시키세요.** 1명이 좋아하면 100명도 좋아할 수 있어요.
-

다음 챕터 미리보기

기획이 끝났으면 이제 만들 차례예요!

Chapter 8에서는 AI에게 기능을 요청하는 방법을 배워요.

- 좋은 프롬프트 작성법
- 기능을 하나씩 추가하는 순서
- 에러가 나도 당황하지 않는 방법

드디어 본격적으로 만드는 시간이에요.

함께 시작해 볼까요?

Chapter 08. 기능 하나씩 추가하기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. AI에게 구체적으로 기능을 요청하는 프롬프트를 작성할 수 있습니다.
 2. 기능을 추가하는 올바른 순서를 이해합니다.
 3. 에러가 발생했을 때 AI를 활용해 해결할 수 있습니다.
-

1. 좋은 프롬프트 작성법

프롬프트란?

프롬프트란 AI에게 하는 요청을 말해요.

쉽게 말해서 AI에게 하는 부탁이에요.

이 부탁을 잘 하는 게 핵심이에요.

같은 부탁이라도 방법에 따라 달라요.

"밥 해줘"보다 "김치볶음밥 해줘"가 낫죠.

AI에게도 구체적으로 말해야 해요.

나쁜 프롬프트 vs 좋은 프롬프트

나쁜 프롬프트의 예를 볼까요?

“로그인 기능 만들어줘.”

이러면 AI가 어떻게 할지 헷갈려요.
이메일 로그인? 소셜 로그인?
비밀번호 찾기도 필요한지 모르죠.
좋은 프롬프트는 이렇게 써요.

“회원가입 기능을 추가해줘.
이메일과 비밀번호로 가입할 수 있게 해줘.
가입 버튼을 누르면 환영 메시지가 나오게 해줘.”

차이가 느껴지시나요?
구체적일수록 좋은 결과가 나와요.
AI도 정확히 뭘 해야 할지 알게 돼요.

좋은 프롬프트의 3가지 규칙

규칙 1: 무엇을 원하는지 명확하게

“예쁘게 만들어줘” (X)
“파란색 버튼을 오른쪽 위에 놓아줘” (O)

규칙 2: 한 번에 하나씩 요청하기

“로그인, 게시판, 채팅 다 만들어줘” (X)
“먼저 로그인 기능을 만들어줘” (O)

규칙 3: 예시를 들어주기

“좋은 디자인으로 해줘” (X)
“네이버처럼 검색창이 가운데에 있게 해줘” (O)

2. 기능 추가 순서

기능 추가 우선순위 매트릭스



중요한 것부터 만드세요

기능을 추가할 때 순서가 중요해요.

가장 핵심적인 기능부터 만드세요.

부가 기능은 나중에 해도 돼요.

맛집 사이트를 예로 들면요.

이런 순서가 좋아요.

1. 맛집 목록 보여주기 (핵심!)
2. 검색 기능 추가하기
3. 별점 남기기 기능
4. 사진 업로드 (나중에)
5. 댓글 기능 (나중에)

한 번에 하나씩

한 기능이 완성되면 다음으로 넘어가세요.

작동하는 것을 확인한 후 다음 기능이에요.

이것이 가장 안전한 방법이에요.

두 가지를 동시에 만들면요.

문제가 생겼을 때 원인을 모르게 돼요.

하나씩 하면 문제 찾기가 쉬워요.

실전 프롬프트 예시

맛집 사이트를 만든다고 가정해 볼게요.

AI에게 이렇게 순서대로 요청하세요.

첫 번째 요청:

“맛집 목록 페이지를 만들어줘.

가게 이름, 주소, 전화번호를 보여줘.

목록은 카드 형태로 보여줘.”

두 번째 요청 (첫 번째 완성 후):

“검색 기능을 추가해줘.

가게 이름으로 검색할 수 있게 해줘.

검색창은 페이지 위쪽에 놓아줘.”

세 번째 요청 (두 번째 완성 후):

“별점 기능을 추가해줘.

1점에서 5점까지 별로 표시해줘.

별을 클릭하면 점수가 저장되게 해줘.”

이렇게 하나씩 요청하는 게 핵심이에요.

3. 에러 해결하기

에러는 당연한 거예요

코딩을 하다 보면 에러가 나오요.

이건 초보만 그런 게 아니에요.

전문 개발자도 매일 에러를 만나요.

에러가 나면 당황하지 마세요.

빨간 글씨가 떠도 괜찮아요.

AI가 고쳐줄 수 있거든요.

에러 메시지를 그대로 보여주세요

에러가 나면 이렇게 하세요.

화면에 뜨는 에러 메시지를 복사하세요.

그리고 AI에게 그대로 보여주세요.

“이런 에러가 났어. 고쳐줘.”

[에러 메시지 붙여넣기]

이러면 AI가 원인을 찾아줘요.

수정된 코드도 만들어 줘요.

내가 에러 내용을 몰라도 괜찮아요.

에러 해결 프롬프트 예시

실제로 이렇게 요청하면 돼요.

“아래 에러가 나왔어. 원인을 알려주고 고쳐줘.”

Error: Cannot read property 'map' of undefined

맛집 목록을 보여주려고 했는데 안 돼."

AI가 친절하게 설명해 줄 거예요.

"이 에러는 데이터가 없어서 생긴 거예요."

이렇게 원인과 해결책을 알려줘요.

잘 안 될 때는 다시 설명하세요

한 번에 안 고쳐질 수도 있어요.

그럴 때는 더 자세히 설명하세요.

** "이전에 알려준 방법이 안 됐어" **라고요.

AI도 대화를 통해 이해가 깊어져요.

몇 번 주고받으면 해결될 거예요.

포기하지 마시고 다시 물어보세요.

실생활 비유: 집 인테리어

새 집에 이사한 상황을 떠올려 보세요.

이사 첫날 모든 방을 꾸밀 수 있나요?

불가능하죠.

현명한 방법은 이렇게 하는 거예요.

먼저 거실부터 꾸며요.

소파를 놓고, TV를 설치해요.

거실이 완성되면 편하게 쉴 수 있죠.

그 다음에 침실을 꾸며요.

침대를 놓고, 커튼을 달아요.

침실이 되면 편히 잘 수 있어요.

그 다음에 주방을 정리해요.

이렇게 한 공간씩 완성하는 거예요.

조금 해하지 않아도 돼요.

웹 서비스도 똑같습니다.

한 기능씩 완성하세요.

하나가 되면 다음으로 넘어가세요.

실제 사례: NomadList 기능 추가 히스토리

Pieter Levels의 NomadList를 다시 볼게요.

이 서비스가 어떻게 성장했는지 살펴봐요.

한 번에 완성된 게 아니에요.

처음에는 도시 목록만 있었어요.

어떤 도시가 있는지 보여주는 것이 전부였어요.

정말 단순한 표 하나였습니다.

그 다음에 날씨 정보를 추가했어요.

"이 도시 날씨는 어때?"라는 질문에 답하려고요.

사용자가 원하는 정보였거든요.

그 다음에 인터넷 속도를 추가했어요.

디지털 노마드에게 인터넷은 필수잖아요.

이것도 사용자 요청이었어요.

그 다음에 커뮤니티 기능을 넣었어요.

사용자들끼리 소통하고 싶어 했거든요.

이렇게 하나씩 기능이 늘어났어요.

핵심이 보이시나요?

한 번에 하나씩 추가했어요.

사용자가 원하는 것을 듣고 추가했어요.

지금 NomadList에는 수십 가지 기능이 있어요.
하지만 처음은 도시 목록 하나였습니다.
여러분도 이렇게 시작하시면 돼요.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. **한 번에 하나씩 요청하세요.** 여러 기능을 동시에 요청하면 결과가 엉망이 돼요.
 2. **구체적으로 설명할수록 좋은 결과가 나와요.** "예쁘게"보다 "파란색 버튼을 위쪽에"가 좋아요.
 3. **에러는 AI에게 그대로 보여주세요.** 에러 메시지를 복사해서 붙여넣기만 하면 돼요.
 4. **작동하면 다음 기능으로 넘어가세요.** 하나가 완성된 후에 다음을 만드는 게 안전해요.
-

다음 챕터 미리보기

기능을 추가하는 방법을 배웠으니요.

Chapter 9에서는 디자인을 다듬는 방법을 배워요.

- 색상과 폰트를 AI에게 요청하는 법
- 깔끔한 레이아웃 만드는 법
- 모바일에서도 잘 보이게 하는 법

기능이 있으면 이제 옷을 입혀야겠죠?
예쁘게 다듬는 방법을 알려드릴게요!

Chapter 09. 디자인 다듬기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. AI에게 색상과 폰트 변경을 요청할 수 있습니다.
 2. 참고 사이트를 보여주며 레이아웃을 개선할 수 있습니다.
 3. 모바일 대응(반응형 디자인)의 개념을 이해합니다.
-

1. 색상과 폰트 바꾸기

색상이 주는 느낌

색상은 서비스의 첫인상이에요.

파란색은 신뢰감을 줘요.

초록색은 자연스러운 느낌이고요.

빨간색은 열정적인 느낌이에요.

내 서비스에 맞는 색을 골라보세요.

건강 관련이면 초록색이 좋겠죠.

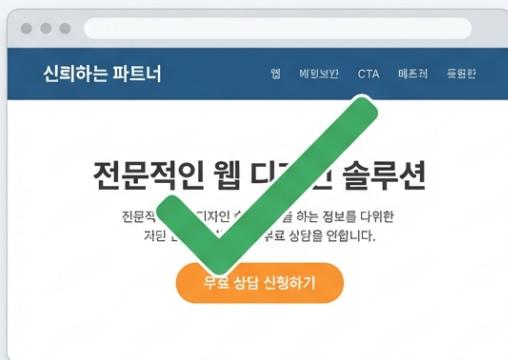
금융 관련이면 파란색이 어울려요.

색상과 폰트가 만드는 차이

변경 전



변경 후



추천 색상 조합

기본 색상
#2C5F8D

보조 색상
#FF9933

배경 색상
#FFFFFF

텍스트 색상: #333333 (다크 그레이)

색상은 3가지 이내로

색상을 너무 많이 쓰면 산만해져요.

3가지 이내로 정하세요.

이것만 지켜도 깔끔해 보여요.

- 주요 색상: 서비스를 대표하는 색 1가지
- 보조 색상: 버튼이나 강조에 쓰는 색 1가지
- 배경 색상: 흰색이나 밝은 회색

이렇게 3가지면 충분합니다.

전문 디자이너도 이 규칙을 따라요.

폰트도 AI에게 맡기세요

폰트란 글씨체를 말해요.

읽기 쉬운 폰트가 가장 좋아요.

예쁜 것보다 잘 읽히는 게 중요해요.

AI에게 이렇게 요청해 보세요.

“디자인을 더 깔끔하게 바꿔줘.
색상은 파란색 계열로 해줘.
폰트는 읽기 쉬운 것으로 바꿔줘.”

AI가 적절한 색상 코드를 골라줘요.
폰트도 가독성 좋은 것을 추천해 줘요.
직접 코드를 수정할 필요가 없어요.

2. 레이아웃 개선하기

레이아웃이란?

레이아웃은 화면의 배치를 말해요.
메뉴가 어디 있고, 내용이 어디 있는지요.
버튼은 어디에 놓을지 정하는 거예요.

신문을 떠올려 보세요.
제목이 위에 있고 기사가 아래에 있죠.
사진은 옆에 배치되어 있고요.
이런 배치가 바로 레이아웃이에요.

참고 사이트를 보여주세요

디자인을 설명하기 어려울 때가 있어요.
그럴 때는 참고 사이트를 알려주세요.
AI에게 이렇게 말하면 돼요.

“이 사이트처럼 만들어줘.
(참고 사이트 주소 붙여넣기)
상단에 메뉴바가 있고
가운데에 큰 이미지가 나오는 형태로 해줘.”

AI가 비슷한 구조로 만들어 줘요.
100% 같지는 않지만 비슷해져요.
말로 설명하는 것보다 훨씬 효과적이에요.

여백의 미학: 밀집 vs 여유



여백을 넉넉히 두세요

초보자가 흔히 하는 실수가 있어요.
화면을 빽빽하게 채우는 거예요.
여백이 있어야 읽기 편해요.

AI에게 이렇게 요청하면 돼요.

“요소들 사이에 여백을 더 넣어줘.
글씨와 이미지 사이를 넓혀줘.
전체적으로 여유 있게 배치해줘.”

여백은 비어 있는 게 아니에요.
읽는 사람을 배려하는 공간이에요.

3. 모바일 대응 (반응형 디자인)

반응형 디자인: 하나의 사이트, 세 가지 화면



반응형이 뭔가요?

요즘 사람들은 스마트폰으로 인터넷을 봐요.
컴퓨터 화면과 스마트폰 화면은 크기가 다르죠.
반응형이란 화면 크기에 맞춰 바뀌는 거예요.

컴퓨터에서는 옆으로 넓게 보이고요.
스마트폰에서는 위아래로 길게 보여요.
같은 내용인데 배치만 달라지는 거예요.

왜 모바일이 중요한가요?

방문자의 70% 이상이 스마트폰을 써요.
컴퓨터에서만 잘 보이면 부족해요.
스마트폰에서 잘 보여야 성공해요.

글씨가 너무 작으면 읽기 힘들죠.
버튼이 너무 작으면 누르기 어렵고요.
모바일에서 불편하면 바로 떠나버려요.

AI에게 반응형을 요청하세요

어렵게 생각할 필요 없어요.
AI에게 한마디만 하면 돼요.

“모바일에서도 잘 보이게 만들어줘.
스마트폰에서 글씨가 잘 읽히게 해줘.
버튼은 손가락으로 누르기 쉽게 해줘.”

AI가 반응형 코드를 만들어 줘요.
내가 코드를 이해할 필요 없어요.
요청만 잘 하면 됩니다.

확인하는 방법

만든 후에 꼭 확인해 보세요.
컴퓨터 브라우저 창의 크기를 줄여보세요.
가로 폭을 좁히면 모바일 화면처럼 돼요.

글씨가 잘 보이나요?
버튼을 쉽게 누를 수 있나요?
이미지가 깨지지 않나요?
이 세 가지만 확인하시면 돼요.

실생활 비유: 옷 입히기

디자인은 옷 입히기와 같아요.
기능이 완성된 서비스는 알몸이에요.
이제 옷을 입혀야 할 차례예요.

그런데 옷도 잘 입어야 하잖아요.
유행을 따르는 것보다 중요한 게 있어요.
깔끔한 기본이 최고예요.

흰 셔츠에 청바지를 입어보세요.
100가지 액세서리보다 나아요.
단정하고 깔끔한 느낌이 들죠.

웹 디자인도 마찬가지예요.
화려한 효과보다 깔끔함이 좋아요.
색상 3가지, 읽기 쉬운 폰트.
이것만으로도 충분히 멋져요.

실제 사례: 심플한 디자인으로 성공한 서비스들

화려한 디자인만 성공할까요?
전혀 아니에요.
오히려 단순한 디자인이 성공하는 경우가 많아요.

Google 검색

구글 첫 화면을 떠올려 보세요.

검색창 하나만 있어요.

나머지는 전부 여백이에요.

그런데 전 세계에서 가장 많이 쓰는 사이트죠.

단순함이 바로 구글의 힘이에요.

사용자가 뭘 해야 할지 바로 알 수 있어요.

Craigslist

Craigslist(크레이그리스트)를 아시나요?

미국의 중고 거래 사이트예요.

디자인이 솔직히 못생겼어요.

텍스트와 링크만 잔뜩 있어요.

예쁘다고 말할 수 없는 사이트예요.

그런데 매달 수십억 명이 방문해요.

이유가 뭘까요?

유용하기 때문이에요.

예쁜 것보다 쓸모 있는 게 중요해요.

NomadList

NomadList는 깔끔한 표 형태예요.

도시 정보가 한눈에 보여요.

화려하지 않지만 정보가 잘 정리되어 있어요.

이 세 사이트의 공통점이 보이시나요?

깔끔하고 유용한 것이 좋은 디자인이에요.

화려함이 아니라 사용하기 편한 것이 핵심이에요.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 깔끔한 디자인이 좋은 디자인이에요. 화려한 것보다 읽기 쉽고 사용하기 편한 게 중요해요.
 2. AI에게 참고 사이트를 보여주세요. 말로 설명하기 어려우면 비슷한 사이트를 알려주는 게 효과적이에요.
 3. 모바일에서도 잘 보이는지 확인하세요. 방문자 대부분이 스마트폰을 사용하니 꼭 점검해야 해요.
 4. 색상은 3가지 이내로 유지하세요. 주요 색, 보조 색, 배경 색만 있으면 충분합니다.
-

다음 챕터 미리보기

디자인까지 다듬었으니 거의 다 됐어요!

Chapter 10에서는 사용자 피드백 받는 방법을 배워요.

- 주변 사람에게 테스트 부탁하는 법
- 피드백을 받아서 개선하는 방법
- 진짜 사용자를 찾는 방법

만든 것을 보여주고 더 좋게 만들어 봐요.

다음 챕터에서 만나요!

Chapter 10. 사용자 피드백 받고 개선하기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. 주변 사람 5명에게 효과적으로 피드백을 받는 방법을 알게 됩니다.
 2. 받은 피드백을 중요/나중에/무시로 분류하는 기준을 세울 수 있습니다.
 3. 우선순위를 정해서 효율적으로 개선하는 과정을 실행할 수 있습니다.
-

1. 주변 5명에게 보여주기

완벽하지 않아도 보여주세요

“아직 완성이 안 됐는데…”

많은 분들이 이렇게 생각합니다.

하지만 100% 완성을 기다리면 영원히 못 보여줍니다.

70%만 되어도 보여주세요.

그래야 방향을 바로잡을 수 있습니다.

누구에게 보여줄까요?

처음 피드백을 받을 5명을 골라보세요.

- 가족 1명 — 솔직한 의견을 줍니다
- 친구 1~2명 — 편하게 말해줍니다
- 동료나 지인 1~2명 — 객관적 시각을 줍니다

이 5명이 여러분의 첫 번째 사용자입니다.
전문가가 아니어도 괜찮습니다.
오히려 일반 사용자의 눈이 더 중요합니다.

피드백을 잘 받는 방법

피드백을 부탁할 때 이렇게 말해보세요.

“내가 만든 서비스인데 한번 써볼래?
솔직하게 말해줘. 안 좋은 점도 괜찮아.”

그리고 이 세 가지를 꼭 물어보세요.

1. “처음 봤을 때 뭔지 바로 알겠어?”
2. “사용하면서 불편한 점이 있었어?”
3. “다시 쓰고 싶은 마음이 들어?”

이 세 질문이면 충분합니다.
너무 많이 물어보면 상대방이 부담스러워합니다.

중요한 태도

피드백을 들을 때 가장 중요한 것이 있습니다.

변명하지 마세요.

"그건 원래 이런 거야"라고 하면 안 됩니다.
사용자가 불편하다고 느꼈다면 그게 사실입니다.

듣고, 적고, 감사하다고 말하세요.
그것으로 충분합니다.

2. 피드백 정리하기: 중요/나중에/무시

피드백이 쌓이면 정리가 필요합니다

5명에게 피드백을 받으면 의견이 다양합니다.

어떤 것은 급하고, 어떤 것은 나중에 해도 됩니다.

어떤 것은 무시해도 됩니다.

세 가지 바구니에 나눠 담아보세요.

바구니 1: 중요 (지금 바로 고치기)

이런 피드백은 바로 반영하세요.

- 2명 이상이 같은 말을 했을 때
- 서비스를 아예 사용할 수 없는 문제일 때
- 처음 화면에서 뭘지 모르겠다고 할 때

예를 들어 보겠습니다.

“버튼을 눌렀는데 아무 반응이 없어요.”

“이게 뭘 하는 서비스인지 모르겠어요.”

이런 건 바로 고쳐야 합니다.

바구니 2: 나중에 (다음에 개선)

지금 당장은 아니지만 기록해두세요.

- 1명만 말한 불편 사항
- “이런 기능도 있으면 좋겠다”는 추가 요청
- 고치면 좋지만 급하지 않은 것들

예를 들면 이런 것입니다.

“색깔을 바꾸면 더 예쁠 것 같아요.”

“알림 기능이 있으면 좋겠어요.”

메모해두고 나중에 하나씩 처리하세요.

바구니 3: 무시 (반영하지 않기)

모든 피드백을 다 반영할 필요는 없습니다.

- 내 서비스의 방향과 맞지 않는 의견
- 너무 개인적인 취향의 문제
- 다른 서비스를 만들어야 하는 수준의 요청

예를 들어 보겠습니다.

“이거 말고 완전히 다른 걸 만들면 안 돼요?”

이런 건 웃으며 넘기면 됩니다.

내 서비스의 핵심을 잃으면 안 됩니다.

AI로 피드백 정리하기

피드백 정리도 AI에게 도움받을 수 있습니다.

“내 서비스에 대한 사용자 피드백을 정리해줘.

아래 의견들을 ‘바로 수정’, ‘나중에’, ‘무시’로 분류해줘.

1. 버튼이 너무 작아서 누르기 어렵다
2. 색깔이 마음에 안 듈다
3. 로그인이 안 된다
4. 게임 기능을 추가해달라”

AI가 깔끔하게 분류해줍니다.

3. 우선순위 정해서 개선하기

한 번에 하나씩

피드백을 분류했으면 이제 개선할 차례입니다.
하지만 한꺼번에 다 고치려고 하면 안 됩니다.

가장 중요한 것 하나부터 시작하세요.

개선 순서 정하기

“중요” 바구니 안에서도 순서가 있습니다.

1. 사용 자체가 불가능한 문제 — 가장 먼저
2. 많은 사람이 말한 문제 — 그다음
3. 쉽게 고칠 수 있는 문제 — 그다음

쉽게 고칠 수 있는 것부터 하는 것도 좋습니다.
작은 성과가 쌓이면 자신감이 생깁니다.

AI에게 개선 요청하기

피드백을 반영할 때도 AI에게 말하면 됩니다.

“사용자들이 버튼이 너무 작다고 해.
모든 버튼 크기를 1.5배로 키워줘.
모바일에서도 잘 눌리게 해줘.”

AI가 코드를 수정해줍니다.
여러분은 결과를 확인하면 됩니다.

개선 후 다시 확인하기

고친 다음에는 다시 보여주세요.

같은 사람에게 보여줘도 좋습니다.

“저번에 말해준 부분 고쳤는데 확인해줄래?”

이 과정을 반복하는 게 **개선의 핵심**입니다.

만들기 → 보여주기 → 듣기 → 고치기.

이 사이클이 서비스를 성장시킵니다.

실생활 비유: 시식 코너

마트에 가면 시식 코너가 있습니다.

떡볶이를 새로 만들었다고 해볼까요?

시식을 나눠줍니다

지나가는 손님에게 한 입씩 드립니다.

그리고 반응을 살핍니다.

“어떠세요? 맛이 괜찮으세요?”

반응을 모읍니다

손님들의 반응이 다양합니다.

- “좀 짜요” — 3명이 말함
- “매운맛도 있으면 좋겠어요” — 1명
- “나는 떡볶이 안 좋아해요” — 1명

핵심만 반영합니다

3명이 짜다고 했으니 소금을 줄입니다.

매운맛은 나중에 고려합니다.

떡볶이를 안 좋아하는 건 어쩔 수 없습니다.

모든 의견을 다 반영할 수는 없습니다.

여러 사람이 공통으로 말하는 것이 핵심입니다.

피드백도 마찬가지입니다.

모든 사용자를 만족시킬 수는 없습니다.

가장 많이 나오는 의견에 집중하세요.

실제 사례: Pieter Levels의 트위터 피드백 전략

만드는 과정을 공개했습니다

Pieter Levels(피터 레벨스)는 독특한 방법을 썼습니다.

만드는 과정을 트위터에 공유한 것입니다.

이것을 “Build in Public”(공개 개발)이라고 합니다.

말 그대로 만드는 과정을 공개하는 방식입니다.

사용자가 직접 의견을 줬습니다

그가 트위터에 이렇게 올렸습니다.

“NomadList에 새 기능을 추가했다.

각 도시의 인터넷 속도를 보여주는 기능이다.

어떻게 생각하세요?”

그러면 사용자들이 바로 답했습니다.

“좋아요! 카페 와이파이 속도도 보여주면 좋겠어요.”
“도시별 비교 기능도 있으면 편할 것 같아요.”

설문조사를 만들 필요가 없었습니다.
트위터가 피드백 창구가 된 것입니다.

NomadList가 진화했습니다

이 피드백들을 반영하면서 NomadList는 점점 좋아졌습니다.

처음에는 단순한 도시 목록이었습니다.
사용자 피드백을 받으면서 이렇게 발전했습니다.

- 도시별 생활비 정보 추가
- 인터넷 속도 정보 추가
- 날씨, 안전도 정보 추가
- 사용자끼리 채팅 기능 추가

모두 사용자가 요청한 것들이었습니다.
Pieter는 듣고, 분류하고, 반영했을 뿐입니다.

우리도 할 수 있습니다

꼭 트위터 팔로워가 많을 필요는 없습니다.
처음에는 주변 5명이면 충분합니다.

중요한 건 듣는 자세입니다.
사용자의 목소리가 서비스를 키웁니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 가까운 5명에게 먼저 보여주세요. 완벽하지 않아도 괜찮습니다.
 2. 모든 피드백을 반영할 필요는 없습니다. 중요/나중에/무시로 분류하세요.
 3. 여러 사람이 공통으로 말하는 의견을 우선적으로 반영하세요.
 4. 피드백은 비판이 아니라 성장의 연료입니다. 감사하게 받아들이세요.
 5. 만들기 → 보여주기 → 듣기 → 고치기. 이 사이클을 반복하는 것이 핵심입니다.
-

다음 챕터 미리보기

다음은 Part 3 종합 실습입니다.

- 지금까지 배운 프로젝트 기획, 제작, 배포, 피드백을 한 번에 실습합니다
- 실제로 서비스를 만들고 주변 사람에게 보여주는 연습을 합니다
- Part 3의 모든 내용을 직접 경험으로 만들어 봅니다

배운 것을 직접 해봐야 내 것이 됩니다.

종합 실습에서 실전 경험을 쌓아보겠습니다.

Part 3 마무리: 나만의 앱 아이디어 구체화

Part 3 핵심 요약

네 개의 챕터를 통해 이런 것들을 배웠습니다.

Chapter 7 — 아이디어를 구체적인 기획으로 바꾸기

- 막연한 아이디어를 구체적인 문장으로 정리하는 법을 배웠습니다.
- AI에게 기획을 도와달라고 부탁할 수 있습니다.
- 처음부터 완벽한 기획은 없습니다. [작게 시작](#)하면 됩니다.

Chapter 8 — AI에게 기능 추가 요청하기

- 기능은 한 번에 하나씩 요청하는 것이 좋습니다.
- AI에게 구체적으로 설명 할수록 결과가 좋아집니다.
- 원하는 대로 안 나오면 다시 부탁하면 됩니다.

Chapter 9 — 보기 좋게 디자인 다듬기

- 디자인 감각이 없어도 AI가 도와줍니다.
- “깔끔하게”, “따뜻하게” 같은 [느낌을 말로](#) 전달하면 됩니다.
- 색상, 글꼴, 배치를 AI에게 자유롭게 요청할 수 있습니다.

Chapter 10 — 다른 사람에게 보여주고 피드백 받기

- 완성 전에 보여줘야 더 좋은 결과가 나옵니다.
 - 피드백은 선물입니다. 부정적인 의견도 소중합니다.
 - 1명에게 보여주는 것만으로도 충분합니다.
-

실습 과제: 내 프로젝트 기획서 1장 작성하기

아이디어를 머릿속에만 두지 마세요.

종이 한 장에 정리하면 훨씬 명확해집니다.

나의 프로젝트 기획서

아래 빈칸을 채워 보세요.

어렵게 생각하지 않으셔도 됩니다.

프로젝트 이름: _____

(예: “우리 동네 산책 친구”, “오늘의 건강 기록장”)

해결하고 싶은 문제: _____

(예: “혼자 산책하면 심심한데, 같이 걸을 사람을 찾기 어렵다”)

핵심 기능 3가지:

1. _____

2. _____

3. _____

(예: 산책 일정 등록, 근처 산책 친구 찾기, 채팅하기)

타겟 사용자: _____

(예: “50대 이상 건강 걷기를 좋아하는 분들”)

디자인 컨셉: _____

(예: “따뜻한 느낌, 큰 글씨, 심플한 화면”)

잘 안 떠오르시나요? AI에게 도움을 받아 보세요

Claude에게 이렇게 말해 보세요.

“나는 50대이고, _____에 관심이 많아.
이 분야에서 사람들이 불편해하는 것이 뭐가 있을까?
간단한 앱 아이디어를 3개만 제안해 줘.”

AI가 제안한 아이디어를 보고 마음에 드는 것을 고르면 됩니다.

기획서 예시

항목	내용
프로젝트 이름	오늘의 텃밭 일기
해결하는 문제	텃밭 작물 관리를 체계적으로 하고 싶다
핵심 기능 1	작물별 물주기, 비료주기 기록
핵심 기능 2	성장 사진 올리기
핵심 기능 3	다른 텃밭 농부와 팁 공유
타겟 사용자	주말농장, 옥상 텃밭을 가꾸는 5060 세대
디자인 컨셉	자연 느낌, 초록색 톤, 큰 글씨

이 기획서가 앞으로 만들 프로젝트의 설계도가 됩니다.

자가 진단 체크리스트

Part 3를 잘 마쳤는지 스스로 점검해 보세요.

아래 항목을 읽고 체크해 보세요.

- [] 나만의 아이디어를 구체화했다.
→ 기획서 빙칸을 채웠다면 성공입니다
- [] MVP(최소 기능 제품)에 들어갈 기능을 정했다.
→ 3개 이하로 줄이는 것이 핵심입니다
- [] AI에게 기능을 요청해 봤다.
→ "이런 기능을 만들어 줘"라고 말해 보셨나요?
- [] 디자인을 수정해 봤다.
→ 색상이든 글씨 크기든, 한 가지라도 바꿔 봤다면 충분합니다
- [] 1명 이상에게 내 아이디어를 보여줬다.
→ 말로 설명하든, 화면을 보여주든 상관없습니다

결과 확인

- 5개 모두 체크: 대단합니다! 이미 기획자이자 개발자입니다.
- 3~4개 체크: 훌륭합니다. 부족한 부분은 다음에 채우면 됩니다.
- 1~2개 체크: 괜찮습니다. 기획서부터 천천히 작성해 보세요.
- 0개 체크: 걱정 마세요. Chapter 7을 다시 읽으면서 시작해 보세요.

당신도 할 수 있습니다

혹시 이런 생각이 드시나요?

“내 아이디어가 별로인 것 같은데…”

“이런 거 누가 쓸까?”

“다른 사람들은 더 대단한 걸 만들던데…”

한 가지 이야기를 해드리겠습니다.

Pieter Levels는 12개 프로젝트 중 대부분 실패했습니다

Pieter Levels를 기억하시나요?

연 38억 원을 버는 1인 개발자입니다.

그가 처음부터 성공한 것은 아닙니다.

12개월 동안 12개의 프로젝트를 만들었습니다.

그중 대부분은 실패했습니다.

아무도 관심을 갖지 않았습니다.

하지만 몇 개가 성공했고, 그것으로 충분했습니다.

“12개를 만들면 1~2개는 된다.

나머지 10개는 배움이다.”

이것이 Pieter Levels의 철학입니다.

여러분의 첫 아이디어도 소중합니다

성공하든 실패하든 상관없습니다.

첫 기획서를 쓴 것 자체가 의미 있는 일입니다.

안 되면 다음 아이디어를 시도하면 됩니다.

AI가 있으니 다시 만드는 것은 어렵지 않습니다.

중요한 것은 멈추지 않는 것입니다.

Part 4 미리보기: 수익화 시작하기

Part 3에서는 나만의 프로젝트를 기획하고 만들어 봤습니다.

Part 4에서는 **돈을 벌 수 있는 구조**를 만들어 봅니다.

- Chapter 11: 수익 모델 정하기 (구독, 일회성, 광고 등)
- Chapter 12: 첫 수익 만들기 (결제 연동)
- Chapter 13: 마케팅으로 사람들에게 알리기
- Chapter 14: 사업을 확장하는 방법

Part 4를 마치면 **실제로 수익을 만들 수 있는 계획**이 완성됩니다.

취미가 사업이 되는 순간을 함께 경험해 봅시다.

준비되셨나요?

다음 장을 넘겨 주세요.

Chapter 11. 수익화 전략 세우기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. 수익 모델의 4가지 종류(광고/구독/일회성/프리미엄)를 이해합니다.
 2. 내 프로젝트에 어떤 모델이 맞는지 판단할 수 있습니다.
 3. 처음 가격을 정하는 기준을 세울 수 있습니다.
-

1. 수익 모델의 종류

돈을 버는 방법은 여러 가지입니다

서비스를 만들었으면 이제 돈을 벌 차례입니다.

"어떻게 돈을 받을까?"가 수익 모델입니다.

크게 네 가지가 있습니다.

1인 사업가의 4가지 수익 모델



광고 수익

블로그/웹사이트 방문자 → 광고 수익

월 예상: 30~100만원



디지털 상품

전자책, 템플릿, 온라인 강의 판매

월 예상: 50~300만원



구독 서비스

월정액 서비스, 멤버십

월 예상: 100~500만원



외주/컨설팅

바이브 코딩 대행, 기술 컨설팅

월 예상: 200~1000만원

모델 1: 광고 수익

서비스에 광고를 붙이는 방식입니다.

- 사용자는 **무료**로 서비스를 씁니다
- 광고주가 돈을 냅니다
- 사용자가 **많아야** 의미 있는 수익이 납니다

예를 들면 유튜브가 이 방식입니다.

영상 중간에 광고가 나오죠.

장점: 사용자가 돈을 안 내도 됩니다.

단점: 사용자가 매우 많아야 합니다.

1인 사업 초기에는 추천하지 않습니다.

사용자가 적으면 수익도 거의 없습니다.

모델 2: 구독 수익

매달 일정 금액을 받는 방식입니다.

- 사용자가 **매월 또는 매년** 결제합니다
- 안정적인 수입이 들어옵니다
- 넷플릭스, 멜론이 이 방식입니다

장점: 수입이 예측 가능합니다.

단점: 매달 가치를 제공해야 합니다.

1인 사업에 가장 추천하는 모델입니다.

사용자 10명만 있어도 수익이 시작됩니다.

모델 3: 일회성 판매

한 번 결제하면 끝나는 방식입니다.

- 전자책, 템플릿, 도구 등이 해당됩니다
- 한 번 만들어 놓으면 **계속 팔 수** 있습니다
- 쿠팡에서 물건 사는 것과 같습니다

장점: 만들어 놓으면 추가 노력이 적습니다.

단점: 매달 새로 팔아야 합니다.

모델 4: 프리미엄(무료+유료)

기본은 무료, 추가 기능은 유료인 방식입니다.

- **무료** 버전으로 사용자를 모읍니다
- 더 좋은 기능은 **유료**로 제공합니다
- 카카오톡이 이 방식입니다

기본 메시지는 무료입니다.

이모티콘이나 선물하기는 유료입니다.

장점: 사용자를 모으기 쉽습니다.

단점: 무료에서 유료 전환이 어렵습니다.

한눈에 비교하기

모델	수익 안정성	시작 난이도	추천도
광고	낮음	쉬움	★★
구독	높음	보통	★★★★★
일회성	보통	쉬움	★★★★
프리미엄	보통	어려움	★★★

2. 내 프로젝트에 맞는 모델 찾기

서비스 종류별 추천 모델

어떤 모델이 맞는지 헷갈리시죠?

만든 서비스의 종류에 따라 다릅니다.

도구형 서비스를 만들었다면:

매일 쓰는 서비스입니다.

할 일 관리, 가계부 같은 것들이죠.

→ 구독 모델이 잘 맞습니다.

정보형 서비스를 만들었다면:

특정 정보를 제공하는 서비스입니다.

맛집 추천, 여행 정보 같은 것들이죠.

→ 프리미엄 모델이 잘 맞습니다.

콘텐츠를 만들었다면:

전자책, 강의, 템플릿 같은 것들입니다.

→ 일회성 판매가 잘 맞습니다.

모델은 섞어도 됩니다

하나만 고를 필요는 없습니다.

예를 들어 볼까요?

- 기본 기능은 **무료** (사용자 모으기)
- 고급 기능은 **월 구독** (안정 수익)
- 특별 템플릿은 **일회성 판매** (추가 수익)

이렇게 조합하면 더 좋습니다.

처음에는 하나로 시작하고 나중에 추가하세요.

3. 가격 정하기

가장 어려운 질문

“얼마를 받아야 하죠?”

많은 분들이 이 질문에서 막힙니다.

너무 비싸면 안 살 것 같고,

너무 싸면 의미가 없을 것 같습니다.

가격을 정하는 세 가지 기준

첫째, 비슷한 서비스를 참고하세요.

내 서비스와 비슷한 것을 찾아보세요.

그 서비스의 가격이 기준이 됩니다.

비슷한 서비스가 월 5,000원이라면,

처음에는 월 3,000원으로 시작해보세요.

둘째, 처음에는 낮게 시작하세요.

첫 사용자를 모으는 것이 중요합니다.

가격이 낮아야 부담 없이 시작합니다.

나중에 가치가 증명되면 올리면 됩니다.

반대로 내리는 건 더 어렵습니다.

셋째, "커피 한 잔 가격"을 기준으로 하세요.

월 5,000원은 커피 한 잔 값입니다.

이 정도면 대부분 부담 없이 결제합니다.

AI에게 가격 전략 물어보기

가격 정하기가 어려우면 AI에게 물어보세요.

“나는 직장인을 위한 할 일 관리 앱을 만들었어.
비슷한 서비스들의 가격대를 알려줘.
처음 시작할 때 적절한 가격을 추천해줘.”

AI가 시장 조사를 도와줍니다.

실생활 비유: 과일 가게

과일 가게를 하나 연다고 생각해보세요.

같은 과일로도 여러 방식으로 팔 수 있습니다.

낱개 판매 = 일회성 판매

사과를 하나씩 팍니다.

손님이 올 때마다 한 개씩 사갑니다.

서비스로 치면 일회성 결제입니다.

전자책 한 권, 템플릿 한 개 파는 것과 같습니다.

과일 박스 = 프리미엄

기본 박스에는 사과 5개가 들어갑니다.

프리미엄 박스에는 사과 5개 + 배 3개입니다.

기본 기능은 무료로, 좋은 기능은 유료로.

프리미엄 모델과 같은 원리입니다.

매주 배달 = 구독

매주 화요일마다 과일 박스를 배달합니다.

한 달에 한 번 결제하면 매주 받을 수 있습니다.

넷플릭스처럼 매달 결제하는 구독 모델입니다.

가장 안정적인 수입이 됩니다.

같은 과일, 세 가지 팔기 방법.

같은 서비스, 세 가지 수익 모델.

어떻게 파느냐에 따라 수익 구조가 달라집니다.

실제 사례: Pieter Levels의 수익 모델

한 가지가 아닌 다양한 조합

Pieter Levels(피터 레벨스)는 여러 서비스를 운영합니다.

흥미로운 점은 각각 다른 수익 모델을 쓴다는 것입니다.

NomadList – 구독 모델

디지털 노마드를 위한 도시 정보 사이트입니다.

- 기본 정보는 **무료**로 볼 수 있습니다
- 상세 정보와 커뮤니티는 **유료**입니다
- 가격은 **연 50달러**(약 6만 5천원)입니다

연간 구독이라 안정적인 수입이 들어옵니다.

RemoteOK — 채용 공고 게시비

원격 근무 채용 사이트입니다.

- 구직자는 무료로 이용합니다
- 기업이 공고를 올릴 때 300달러 이상 냅니다
- 기업에서 돈을 받는 B2B 모델입니다

사용자에게는 무료, 기업에게는 유료.

양쪽 모두 만족하는 구조입니다.

PhotoAI — 구독 모델

AI로 사진을 생성하는 서비스입니다.

- 월 구독으로 운영됩니다
- 월 수익이 10만 달러 이상입니다
- 혼자서 운영하고 있습니다

핵심 교훈

Pieter Levels는 처음부터 완벽한 수익 모델을 가진 게 아닙니다.

- 처음에는 무료로 시작했습니다
- 사용자가 모이니 유료로 전환했습니다
- 서비스마다 다른 모델을 적용했습니다

수익 모델은 처음부터 정할 필요 없습니다.

사용자를 모으면서 자연스럽게 찾아가면 됩니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 구독형 모델이 1인 사업에 가장 안정적입니다. 매달 예측 가능한 수입이 들어옵니다.
 2. 처음에는 가격을 낮게 시작하세요. 첫 사용자를 모으는 것이 더 중요합니다.
 3. 무료 버전으로 사용자를 모으고, 유료로 전환하는 전략이 효과적입니다.
 4. 수익 모델은 나중에 변경 가능합니다. 처음부터 완벽할 필요 없습니다.
 5. 같은 서비스로도 여러 수익 모델을 조합할 수 있습니다.
-

다음 챕터 미리보기

Chapter 12에서는 실제로 결제 시스템을 연동합니다.

- Stripe, 토스페이먼츠 같은 결제 도구를 알아봅니다
- AI에게 결제 연동을 요청하는 방법을 배웁니다
- 첫 고객을 찾고 첫 수익을 만드는 과정을 경험합니다

전략을 세웠으니 이제 실행할 차례입니다.

다음 챕터에서 첫 수익의 기쁨을 느껴보겠습니다.

Chapter 12. 첫 수익 만들기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

- 결제 시스템을 내 서비스에 연동하는 방법을 알게 됩니다.
 - 첫 번째 고객을 찾는 구체적인 방법을 배웁니다.
 - 가격을 정하고 테스트하는 방법을 이해합니다.
-

1. 결제 시스템 연동하기

돈을 받을 준비부터

서비스를 만들었습니다.

수익 모델도 정했습니다.

이제 실제로 돈을 받을 수 있어야 합니다.

“결제 시스템 연동이 어렵지 않을까?”

걱정하지 마세요.

AI와 함께하면 생각보다 간단합니다.

Stripe: 해외 결제의 표준

Stripe(스트라이프)는 전 세계에서 가장 많이 쓰는 결제 서비스입니다.

특징을 정리하면 이렇습니다.

- 가입이 무료입니다
- 결제가 일어날 때만 수수료를 냅니다

- 신용카드, 체크카드를 받을 수 있습니다
- 해외 고객도 결제할 수 있습니다

수수료는 결제 금액의 약 2.9% + 30센트입니다.

1만 원을 받으면 약 320원 정도가 수수료입니다.

토스페이먼츠: 국내 결제에 강합니다

한국 고객이 주 대상이라면 토스페이먼츠가 좋습니다.

- 카카오페이지, 네이버페이지, 토스를 지원합니다
- 한국 사람들에게 익숙한 결제 방식입니다
- 수수료는 약 2.5~3.5% 수준입니다
- 한국어로 된 안내 문서가 있습니다

AI에게 결제 연동 요청하기

결제 시스템 연동도 AI에게 맡길 수 있습니다.

이렇게 요청해 보세요.

“내 웹사이트에 Stripe 결제 버튼을 추가해줘.
월 9,900원 구독 상품이야.
결제 완료 후 감사 페이지로 이동하게 해줘.”

AI가 필요한 코드를 만들어 줍니다.

복사해서 붙여넣으면 됩니다.

처음에는 간단하게

처음부터 복잡한 결제 시스템을 만들 필요 없습니다.

1단계: 결제 버튼 하나 추가하기

2단계: 결제 완료 페이지 만들기

3단계: 고객에게 서비스 제공하기

이 세 단계면 충분합니다.
자동화나 구독 관리는 나중에 추가하면 됩니다.

2. 첫 고객 찾기

첫 고객 1명이 가장 중요합니다

100명의 고객보다 **첫 1명**이 더 중요합니다.

왜냐하면 첫 고객이 증명해 주기 때문입니다.
“진짜 돈을 내고 쓸 사람이 있구나.”

이 확신이 생기면 모든 것이 달라집니다.

주변 사람부터 시작하세요

첫 고객은 멀리 있지 않습니다.

가족과 친구에게 보여주세요.
“이런 서비스를 만들었는데 한번 봐줄래?”
솔직한 피드백을 받을 수 있습니다.

직장 동료에게 알려주세요.
같은 업종의 불편함을 해결하는 서비스라면
동료가 첫 고객이 될 수 있습니다.

동호회에서 소개하세요.
취미 관련 서비스라면 동호회 회원들이 관심을 보일 겁니다.

온라인에서 찾기

주변에 적합한 사람이 없다면 온라인으로 넓혀보세요.

네이버 카페:

관련 주제의 카페에 가입합니다.

활동을 하면서 자연스럽게 서비스를 소개합니다.

페이스북 그룹:

관심사가 비슷한 그룹에서 활동합니다.

"이런 걸 만들어 봤는데 어떠세요?"라고 물어봅니다.

인스타그램:

서비스와 관련된 해시태그를 검색합니다.

관심 있는 사람들에게 DM을 보냅니다.

무료로 먼저 써보게 하세요

첫 고객에게는 **무료 체험**을 제공하세요.

“한 달 동안 무료로 써보시겠어요?”

무료라면 부담 없이 시작할 수 있습니다.

서비스가 좋으면 유료로 전환됩니다.

이것을 **프리 트라이얼**(무료 체험)이라고 합니다.

대부분의 성공한 서비스가 이 방법을 씁니다.

3. 가격 테스트하기

가격 정하기가 가장 어렵습니다

“얼마를 받아야 하지?”

이 질문에 정답은 없습니다.

하지만 **시작점**은 있습니다.

경쟁 서비스를 살펴보세요

비슷한 서비스가 얼마를 받는지 찾아보세요.

- 비슷한 서비스가 월 1만 원이라면
- 내 서비스는 월 5,000~8,000원으로 시작합니다

처음에는 낮게 시작하는 게 좋습니다.

사용자가 모이면 가격을 올릴 수 있습니다.

세 가지 가격 만들기

가격을 하나만 두지 마세요.

세 가지 옵션을 만드세요.

플랜	가격	내용
기본	무료	핵심 기능만
프로	월 9,900원	모든 기능
프리미엄	월 19,900원	모든 기능 + 우선 지원

대부분의 사람은 가운데 옵션을 선택합니다.

이것을 앵커링 효과라고 합니다.

비교 대상이 있으면 가운데가 적당해 보이는 심리입니다.

가격은 바꿀 수 있습니다

처음 정한 가격이 영원한 게 아닙니다.

한 달 운영해 보고 반응을 봅니다.

"너무 비싸다"는 피드백이 오면 낮춥니다.

"이 가격에 이 기능이?"라는 반응이면 올릴 수 있습니다.

가격은 실험입니다.

데이터를 보고 조정하면 됩니다.

실생활 비유: 시장에서 첫 손님 맞이하기

노점을 열었습니다

시장에서 떡볶이 노점을 열었다고 상상해 보세요.

장소를 정했습니다.

재료를 준비했습니다.

떡볶이를 만들었습니다.

이제 손님을 기다립니다.

첫 손님이 가장 떨립니다

한 시간이 지나도 손님이 없습니다.

“맛이 없나? 가격이 비싼가?”

불안해집니다.

그때 한 분이 다가옵니다.

“떡볶이 하나 주세요.”

이 한마디가 모든 것을 바꿉니다.

떡볶이를 내놓습니다.

그분이 맛있게 드십니다.

“맛있네요!”

이 한마디에 자신감이 생깁니다.

“내 떡볶이를 좋아하는 사람이 있구나!”

첫 손님이 단골이 됩니다

그분이 다음 날 또 옵니다.

친구를 데려옵니다.

“여기 떡볶이 맛있어.”

입소문이 시작됩니다.

한 명이 두 명이 되고, 두 명이 다섯 명이 됩니다.

디지털 사업도 마찬가지입니다.

첫 고객 1명을 정성껏 대하세요.

그 1명이 다른 사람에게 알려줍니다.

이것이 성장의 시작입니다.

실제 사례: Marc Lou의 첫 100달러

27번 실패 후의 도전

Marc Lou의 이야기를 다시 꺼내보겠습니다.

Chapter 2에서 그가 27번 실패했다는 이야기를 했습니다.

28번째 제품을 만들었을 때도 처음에는 반응이 없었습니다.

하지만 이번에는 뭔가 달랐습니다.

첫 100달러의 의미

28번째 제품에서 첫 결제가 들어왔습니다.

100달러.

약 13만 원입니다.

금액 자체는 크지 않습니다.

하지만 Marc Lou에게는 세상이 바뀐 순간이었습니다.

“내가 만든 것에 누군가 돈을 냈다.

이건 진짜다. 이건 사업이다.”

27번의 실패 동안 한 번도 못 느낀 감정이었습니다.

100달러가 1000달러가 되기까지

첫 100달러 이후 Marc Lou가 한 일은 간단했습니다.

첫째, 고객에게 물어봤습니다.

“왜 구매하셨어요?”

“어떤 기능이 가장 좋았어요?”

둘째, 피드백을 반영했습니다.

고객이 원하는 기능을 추가했습니다.

불편한 부분을 개선했습니다.

셋째, 과정을 공개했습니다.

트위터에 수익을 공개했습니다.

다른 사람들이 관심을 가졌습니다.

첫 달 100달러.

두 번째 달 500달러.

세 번째 달 1,000달러.

시작은 작았지만 성장은 빨랐습니다.

여러분의 첫 100달러

Marc Lou도 처음에는 아무도 몰랐습니다.

특별한 인맥도 없었습니다.

하지만 좋은 제품을 만들었습니다.

첫 고객을 정성껏 대했습니다.

피드백을 들었습니다.

여러분도 똑같이 할 수 있습니다.

첫 1명의 고객을 찾으세요.

그 1명에게 최선을 다하세요.

그것이 첫 수익의 시작입니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 결제 시스템은 AI가 도와줍니다. Stripe나 토스페이먼츠를 연동하는 것도 AI에게 요청할 수 있습니다.
 2. 첫 고객 1명이 가장 중요합니다. 100명보다 정성을 다한 1명이 사업의 시작입니다.
 3. 가격은 실험입니다. 처음부터 완벽한 가격은 없습니다. 시작하고 조정하세요.
 4. 첫 수익의 기쁨이 동력이 됩니다. 작은 금액이라도 직접 번 돈은 큰 자신감을 줍니다.
-

다음 챕터 미리보기

Chapter 13에서는 마케팅의 기초를 배웁니다.

- SNS를 활용해 서비스 알리기
- 콘텐츠로 자연스럽게 홍보하기
- 커뮤니티에서 신뢰 쌓기

광고비 없이도 사람들에게 알릴 수 있습니다.

어떻게 하는지 다음 챕터에서 알아보겠습니다!

Chapter 13. 마케팅의 기초

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. SNS를 활용해 서비스를 알리는 방법을 알게 됩니다.
 2. 콘텐츠 마케팅이 무엇이고 어떻게 하는지 배웁니다.
 3. 커뮤니티 참여로 신뢰를 쌓는 방법을 이해합니다.
-

1. SNS 활용하기

광고비 없이 알리는 방법

서비스를 만들었는데 아무도 모릅니다.

“어떻게 알리지?”

TV 광고를 낼 수는 없습니다.

전단지를 돌릴 수도 없습니다.

하지만 SNS가 있습니다.

무료입니다.

누구나 할 수 있습니다.

어떤 SNS를 쓸까요?

모든 SNS를 다 할 필요 없습니다.

하나만 잘하면 됩니다.

인스타그램: 사진, 짧은 영상 위주

→ 음식, 여행, 운동, 라이프스타일 서비스에 좋습니다.

네이버 블로그: 긴 글 위주

→ 정보 제공, 교육, 건강 서비스에 좋습니다.

유튜브: 영상 위주

→ 사용법 설명, 후기, 비교 분석에 좋습니다.

트위터(X): 짧은 글 위주

→ 개발 과정 공유, 해외 고객 대상 서비스에 좋습니다.

내 고객이 어디에 있는지 생각해 보세요.

고객이 있는 곳에서 활동하세요.

무엇을 올릴까요?

“올릴 게 없는데요?”

아닙니다.

올릴 것은 많습니다.

만드는 과정을 보여주세요.

“오늘 새로운 기능을 추가했습니다.”

“사용자 피드백을 반영해서 이렇게 바꿨습니다.”

도움이 되는 정보를 공유하세요.

“등산할 때 물 마시는 타이밍 알려드립니다.”

“가계부 쉽게 쓰는 방법 3가지”

사용자 후기를 나누세요.

“이런 감사한 메시지를 받았습니다.”

“이렇게 활용하고 계신 분이 있네요.”

꾸준함이 비결입니다

일주일에 2~3개 정도면 충분합니다.

매일 올릴 필요 없습니다.

하지만 **꾸준히** 올려야 합니다.

한 달 열심히 하고 두 달 쉬면 효과가 없습니다.

일주일에 2개씩, 6개월이면 50개입니다.

50개의 게시물은 **든든한 자산**이 됩니다.

2. 콘텐츠 마케팅

도움이 되면 마케팅이 됩니다

"마케팅"이라고 하면 뭐가 떠오르시나요?

화려한 광고?

“지금 바로 구매하세요!” 같은 문구?

콘텐츠 마케팅은 다릅니다.

사람들에게 **도움이 되는 글**을 쓰는 겁니다.

도움을 받은 사람은 자연스럽게 관심을 갖습니다.

콘텐츠 마케팅의 예시

등산 코스 추천 서비스를 만들었다면 이런 글을 쓸 수 있습니다.

블로그 글:

“초보자도 쉽게 오를 수 있는 서울 근교 등산 코스 5선”

“등산 전 꼭 해야 할 스트레칭 5가지”

“등산 배낭 싸는 법: 꼭챙겨야 할 10가지”

글을 읽은 사람이 생각합니다.

“이 사람이 만든 서비스도 좋겠는데?”

이것이 콘텐츠 마케팅입니다.

직접 “사세요!”라고 하지 않아도 됩니다.

도움을 주면 사람들이 알아서 찾아옵니다.

AI로 콘텐츠 만들기

글쓰기가 어렵다면 AI를 활용하세요.

Claude에게 이렇게 요청합니다.

“등산 초보자를 위한 블로그 글을 써줘.

주제는 '서울 근교 쉬운 등산 코스 5곳'이야.

각 코스마다 난이도, 소요 시간, 특징을 넣어줘.

친근한 말투로 써줘.”

AI가 초안을 만들어 줍니다.

여러분의 경험을 추가하면 됩니다.

내 경험이 들어가면 특별한 콘텐츠가 됩니다.

AI만으로는 못 쓰는 진짜 이야기가 있기 때문입니다.

검색에 잘 걸리게 하기

좋은 글을 써도 아무도 안 보면 소용없습니다.

사람들이 검색할 만한 제목을 쓰세요.

나쁜 예: “오늘의 등산 일기”

좋은 예: “초보자 추천 등산 코스 서울 근교 5곳”

사람들은 “초보자 등산 코스”로 검색합니다.

“오늘의 등산 일기”로는 검색하지 않습니다.

사람들이 검색할 법한 단어를 제목에 넣으세요.

이것을 SEO(검색 엔진 최적화)라고 합니다.

3. 커뮤니티 참여하기

혼자 하면 외롭습니다

1인 사업은 혼자 하는 일입니다.

하지만 혼자 있을 필요는 없습니다.

비슷한 일을 하는 사람들이 모인 곳이 있습니다.

커뮤니티입니다.

어디에 참여할까요?

한국 커뮤니티:

- [네이버 카페](#): “1인 창업”, “사이드 프로젝트” 검색
- [오픈카카오톡 방](#): “바이브 코딩”, “AI 활용” 검색
- [디스코드 서버](#): “인디해커”, “개발자 모임” 검색

해외 커뮤니티:

- [Indie Hackers](#): 1인 사업가들의 모임
- [Product Hunt](#): 새 제품을 소개하는 곳
- [Reddit](#): 관심 분야의 서브레딧

커뮤니티에서 하면 좋은 것

첫째, 나의 진행 상황을 공유하세요.

“이번 주에 이런 기능을 추가했습니다.”
“사용자가 10명이 되었습니다!”
“이런 문제에 막혔는데 조언 부탁드립니다.”

진솔하게 공유하면 응원해주는 사람이 생깁니다.

둘째, 다른 사람을 도와주세요.

누군가 질문을 올리면 답해주세요.
내가 아는 것을 나누세요.
도움을 주면 신뢰가 쌓입니다.

셋째, 피드백을 구하세요.

“이 디자인 어떤가요?”
“가격을 이렇게 생각하는데 적당할까요?”

커뮤니티의 피드백은 [무료 컨설팅](#)과 같습니다.

서로 돋는 관계

커뮤니티는 일방적인 곳이 아닙니다.
주는 만큼 받는 곳입니다.

내 서비스를 소개하기만 하면 안 됩니다.
다른 사람의 서비스도 써보고 피드백을 주세요.
서로 응원하고 도와주세요.

이런 관계가 쌓이면 든든한 네트워크가 됩니다.
혼자서는 못 하는 일도 함께하면 할 수 있습니다.

실생활 비유: 동네 입소문

새로 이사 온 가게

동네에 새 빵집이 생겼다고 상상해 보세요.

아무도 모르는 빵집입니다.

간판을 봐도 “맛있을까?” 고민됩니다.

입소문이 시작되는 순간

어떤 아주머니가 들어갑니다.

빵을 사먹어 봅니다.

“어머, 여기 빵 진짜 맛있네!”

그 아주머니가 이웃에게 말합니다.

이웃이 또 다른 이웃에게 말합니다.

일주일 뒤, 빵집에 줄이 섭니다.

입소문의 비결

그 빵집은 광고를 하지 않았습니다.

전단지를 뿌리지 않았습니다.

빵이 맛있었습니다.

사장님 친절했습니다.

이 두 가지가 입소문의 비결입니다.

디지털 사업도 마찬가지입니다.

서비스가 좋으면 사람들이 알아서 알려줍니다.

진심으로 대하면 사람들이 응원해 줍니다.

화려한 마케팅 기술보다 진심이 먼저입니다.

실제 사례: Marc Lou의 트위터 마케팅

만드는 과정을 다 보여줬습니다

Marc Lou의 마케팅 전략은 단순했습니다.

“만드는 과정을 전부 공개한다.”

그는 트위터에 매일 글을 올렸습니다.

“오늘 새 프로젝트를 시작했다.”

“로그인 기능을 만드는 데 3시간 걸렸다.”

“첫 사용자가 가입했다! 기쁘다!”

“오늘 수익: 0달러. 하지만 괜찮다.”

성공만 보여준 게 아닙니다.

실패도, 고민도, 기쁨도 다 보여줬습니다.

사람들이 응원하기 시작했습니다

진솔한 이야기에 사람들이 반응했습니다.

“파이팅!”

“나도 이렇게 해봐야겠다.”

“대단하다, 응원한다.”

팔로워가 늘어났습니다.

팔로워가 고객이 되었습니다.

유튜브로 확장했습니다

트위터에서 어느 정도 팔로워가 모이자

Marc Lou는 유튜브도 시작했습니다.

“24시간 안에 SaaS 만들기” 같은 영상을 올렸습니다.

실시간으로 서비스를 만드는 과정을 보여줬습니다.

영상 하나가 수십만 뷰를 기록했습니다.
유튜브를 본 사람들이 그의 서비스를 구매했습니다.

핵심은 ‘진짜 이야기’

Marc Lou의 마케팅에 특별한 기술은 없었습니다.
비싼 광고도 없었습니다.
있었던 건 솔직한 이야기뿐이었습니다.
여러분도 마찬가지입니다.
바이브 코딩으로 서비스를 만드는 과정 자체가 콘텐츠입니다.

“50대에 처음으로 웹사이트를 만들어 봤습니다.”
“AI에게 요청해서 이런 결과물이 나왔습니다.”
“첫 사용자가 ‘편하다’고 했을 때의 기분.”

이런 이야기가 가장 강력한 마케팅입니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. **만드는 과정을 공유하면 마케팅이 됩니다.** 완성된 것만 보여줄 필요 없습니다.
과정 자체가 콘텐츠입니다.
2. **SNS 하나를 꾸준히 하세요.** 모든 곳에서 활동하려 하지 마세요. 하나만 제대로
하면 됩니다.
3. **도움이 되는 콘텐츠가 최고의 마케팅입니다.** "사세요!"보다 "이게 도움이
됩니다"가 더 효과적입니다.
4. **커뮤니티에서 함께 성장하세요.** 혼자 하지 마세요. 서로 돋는 관계가 큰 힘이
됩니다.

다음 챕터 미리보기

Chapter 14에서는 수익 확장하기를 배웁니다.

- 두 번째 제품 만들기
- 프리미엄 기능 추가하기
- 수익을 다각화하는 방법

첫 수익을 냈다면 이제 키울 차례입니다.

어떻게 성장시킬 수 있는지 알아보겠습니다!

Chapter 14. 수익 확장하기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. 첫 번째 제품의 경험을 활용해 **두 번째 제품**을 만드는 방법을 알게 됩니다.
 2. 기존 제품에 **프리미엄 기능**을 추가해 수익을 높이는 방법을 이해합니다.
 3. **수익 다각화**가 왜 중요한지, 어떻게 하는지 설명할 수 있습니다.
-

1. 두 번째 제품 만들기

첫 제품이 선생님입니다

첫 번째 제품을 만들면서 많이 배우셨을 겁니다.

사용자가 뭘 원하는지 알게 되었습니다.

어떤 기능이 인기 있는지도 보였을 겁니다.

이 경험이 바로 두 번째 제품의 씨앗입니다.

첫 제품을 운영하다 보면 이런 생각이 듭니다.

“이 기능을 따로 떼서 만들면 어떨까?”

“다른 분야에도 이걸 적용할 수 있겠는데?”

그 생각이 두 번째 제품의 시작입니다.

두 번째는 훨씬 쉽습니다

첫 제품을 만들 때를 떠올려 보세요.

모든 게 낯설었습니다.

AI에게 뭘 물어봐야 할지도 몰랐습니다.

두 번째는 다릅니다.

- 어떤 도구를 쓸지 이미 알고 있습니다
- AI에게 지시하는 방법도 익숙합니다
- 사용자가 뭘 원하는지 감이 있습니다
- 결제 시스템도 이미 만들어 봤습니다

첫 제품에 3개월 걸렸다면, 두 번째는 1개월이면 됩니다.

어떤 제품을 만들까?

두 번째 제품의 아이디어는 멀리 있지 않습니다.

방법 1: 기존 사용자에게 물어보기

“또 어떤 기능이 있으면 좋겠어요?”

사용자들이 직접 답을 줄 때가 많습니다.

방법 2: 첫 제품의 일부를 독립시키기

건강 식단 서비스를 만들었다면?

식재료 가격 비교 기능만 떼어낼 수 있습니다.

그것만으로도 하나의 제품이 됩니다.

방법 3: 비슷한 문제를 다른 분야에서 풀기

동네 맛집 추천 서비스를 만들었다면?

동네 병원 추천, 학원 추천도 가능합니다.

구조는 같고 내용만 바뀝니다.

2. 기존 제품 업그레이드

프리미엄 기능 추가하기

새 제품을 만들지 않아도 됩니다.

기존 제품을 업그레이드하는 것도 좋은 방법입니다.

무료로 쓰던 기능 위에 프리미엄을 얹는 겁니다.

무료 기능	프리미엄 기능
하루 3번 검색	무제한 검색
기본 보고서	상세 분석 보고서
광고 포함	광고 제거
이메일 지원	실시간 채팅 지원

이런 구조를 **프리미엄 모델**이라고 합니다.

기본은 무료, 더 좋은 건 유료인 구조입니다.

AI에게 프리미엄 기능 만들기를 부탁하세요

AI에게 이렇게 말하면 됩니다.

“지금 서비스에 프리미엄 구독 기능을 추가해 줘.
월 9,900원 구독자만 쓸 수 있는 기능이야.
무제한 검색과 상세 보고서를 제공해 줘.”

AI가 결제 연동부터 기능 분리까지 도와줍니다.

수익 다각화의 힘

수입원이 하나뿐이면 위험합니다.

그 하나가 막히면 수입이 0이 됩니다.

여러 곳에서 돈이 들어오면 안전합니다.

수익 다각화 방법을 정리해 보겠습니다.

- **구독료**: 매달 정기적으로 들어오는 돈
- **일회성 결제**: 특별 기능이나 콘텐츠 판매
- **광고 수입**: 사용자가 많아지면 가능
- **제휴 수수료**: 다른 서비스를 추천하고 수수료 받기
- **컨설팅**: 경험을 바탕으로 다른 사람 도와주기

하나의 제품에서도 여러 수입원을 만들 수 있습니다.

실생활 비유: 과수원 키우기

수익 확장을 과수원에 비유해 보겠습니다.

사과나무만 심으면 위험합니다

올해 사과 농사가 잘되었습니다.

기뻐서 사과나무만 더 심었습니다.

그런데 다음 해에 태풍이 왔습니다.

사과가 다 떨어졌습니다.

수입이 0이 되었습니다.

여러 과일을 심으면 안전합니다

현명한 농부는 다르게 합니다.

사과나무도 심고, 배나무도 심습니다.

감나무, 포도나무도 함께 키웁니다.

사과 농사가 안 좋아도 배가 잘됩니다.

포도 수확이 적어도 감이 풍년입니다.

전체 수입은 안정적으로 유지됩니다.

시간이 지나면 나무가 자랍니다

처음 심은 묘목은 작습니다.

열매도 얼마 안 열립니다.

하지만 해가 지날수록 나무가 커집니다.

열매도 점점 많이 열립니다.

디지털 제품도 똑같습니다.

처음엔 사용자가 10명입니다.

1년 뒤에는 100명이 됩니다.

2년 뒤에는 1,000명이 될 수 있습니다.

나무를 여러 그루 심어 두면 됩니다.

시간이 알아서 키워 줍니다.

실제 사례: Pieter Levels의 다중 프로젝트 전략

하나가 아니라 여러 개를 운영합니다

Pieter Levels는 한 가지만 하지 않습니다.

여러 제품을 동시에 운영합니다.

그의 주요 제품들을 살펴보겠습니다.

- NomadList: 디지털 노마드를 위한 도시 정보
- RemoteOK: 원격 근무 채용 사이트
- PhotoAI: AI로 사진을 만드는 서비스
- InteriorAI: AI로 인테리어를 바꿔주는 서비스

이 외에도 여러 작은 프로젝트가 있습니다.

한 제품이 안 되면 다른 제품이 받쳐줍니다

RemoteOK의 수입이 줄어든 달이 있었습니다.

하지만 PhotoAI가 그 달에 폭발적으로 성장했습니다.

NomadList의 성장이 멈춘 시기도 있었습니다.

대신 InteriorAI가 새로운 수입원이 되었습니다.

여러 제품이 서로를 보완합니다.

연 38억의 비밀

Pieter Levels의 연 수입은 약 38억 원입니다.

그 비밀은 의외로 단순합니다.

“여러 개의 작은 수입원”

하나의 대박 제품이 아닙니다.

여러 개의 중소형 제품이 합쳐진 것입니다.

각 제품이 월 수백만 원씩 벌어다 줍니다.

그것들이 모이면 큰 금액이 됩니다.

그리고 중요한 점이 하나 더 있습니다.

모든 제품을 [혼자서](#) 운영합니다.

자동화 덕분에 가능한 일입니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 첫 제품이 안정되면 [두 번째 제품](#)을 시작하세요. 첫 번째보다 훨씬 쉽고 빠릅니다.
2. [기존 사용자에게 새 제품을 제안](#)하면 처음부터 시작하는 것보다 훨씬 효과적입니다.
3. [여러 수입원](#)을 만들면 하나가 안 되어도 전체 수입은 안정적입니다.

4. 자동화로 여러 제품을 관리하는 시간을 줄일 수 있습니다. 혼자서도 충분합니다.

5. 한 번에 다 하려고 하지 마세요. 하나씩 차근차근 늘려가면 됩니다.

다음 챕터 미리보기

이제 Part 4의 마지막 단계입니다.

Part 4 종합 실습에서는 배운 것들을 직접 해봅니다.

- 내 첫 제품의 수익 확장 계획 세우기
- 두 번째 제품 아이디어 구체화하기
- 프리미엄 기능 설계해 보기

수익의 씨앗을 여러 곳에 심어 보겠습니다.

Part 4 마무리: 수익화 계획 세우기

Part 4 핵심 요약

네 개의 챕터를 통해 이런 것들을 배웠습니다.

Chapter 11 – 수익 모델 정하기

- 수익 모델에는 구독, 일회성 결제, 광고 등 여러 가지가 있습니다.
- 처음에는 가장 단순한 모델 하나를 선택하는 것이 좋습니다.
- 무료로 시작하고 나중에 유료로 전환해도 됩니다.

Chapter 12 – 첫 수익 만들기

- 결제 연동은 AI에게 부탁하면 생각보다 쉽습니다.
- Stripe, 토스페이먼츠 같은 도구를 활용할 수 있습니다.
- 첫 1,000원이 가장 어렵고, 가장 의미 있습니다.

Chapter 13 – 마케팅으로 알리기

- 좋은 제품도 알려야 사람들이 찾아옵니다.
- SNS, 블로그, 커뮤니티 등 무료 채널부터 시작하면 됩니다.
- 거창한 광고보다 솔직한 이야기가 더 효과적입니다.

Chapter 14 – 사업 확장하기

- 잘 되는 것에 집중하는 것이 확장의 핵심입니다.
- 한 번에 크게 키우려 하지 마세요. 조금씩 늘려 가면 됩니다.
- 자동화할 수 있는 것은 AI에게 맡기세요.

실습 과제: 내 프로젝트의 수익 모델 캔버스 작성하기

아이디어만으로는 사업이 되지 않습니다.

돈이 어떻게 들어올지 정리해 봐야 합니다.

수익 모델 캔버스

아래 표를 채워 보세요.

처음이니까 대략적으로 적어도 괜찮습니다.

항목	내 계획
제품/서비스 이름	_____
수익 모델	_____ (구독 / 일회성 결제 / 광고 / 기타)
가격	월 _____ 원 또는 건당 _____ 원
타겟 고객	_____
목표 고객 수 (첫 3개월)	_____ 명
월 예상 수익	_____ 원
주요 마케팅 채널	_____

잘 모르겠다면 이렇게 생각해 보세요

[수익 모델 고르기](#)

수익 모델	장점	적합한 경우
월 구독	매달 안정적 수익	계속 쓰는 서비스 (기록 앱, 학습 도구)
일회성 결제	한 번에 목돈	한 번 사면 되는 것 (전자책, 템플릿)
광고	사용자 부담 없음	방문자가 많은 사이트
프리미엄	무료로 모으고 유료로 전환	기본 기능은 무료, 고급 기능은 유료

가격 정하기

처음에는 이렇게 생각하면 쉽습니다.

“커피 한 잔 값(5,000원)부터 시작하자.”

부담 없는 가격으로 시작해서 가치를 증명한 후에 올리면 됩니다.

수익 모델 캔버스 예시

항목	내용
제품/서비스 이름	오늘의 텃밭 일기
수익 모델	프리미엄 (기본 무료, 고급 기능 유료)
가격	월 4,900원
타겟 고객	주말농장을 가꾸는 5060 세대
목표 고객 수	첫 3개월에 유료 회원 20명
월 예상 수익	98,000원
주요 마케팅 채널	네이버 카페 (텃밭/주말농장 관련)

월 10만 원이 작아 보일 수 있습니다.

하지만 이것이 시작입니다.

10만 원이 100만 원이 되고, 100만 원이 1,000만 원이 됩니다.

| 자가 진단 체크리스트

Part 4를 잘 마쳤는지 스스로 점검해 보세요.

아래 항목을 읽고 체크해 보세요.

- [] 수익 모델을 하나 정했다.
→ 구독, 일회성, 광고, 프리미엄 중 하나를 골랐나요?
- [] 가격을 정했다.
→ 대략적인 금액이라도 괜찮습니다
- [] 결제 연동 방법을 알게 되었다.
→ Stripe, 토스페이먼츠 등의 이름을 들어 봤다면 충분합니다
- [] 마케팅 채널을 하나 이상 정했다.
→ 블로그, SNS, 카페 등 어디서 알릴지 생각해 봤나요?
- [] 첫 고객 후보 3명을 떠올릴 수 있다.
→ 가족, 친구, 지인 중에 내 서비스를 쓸 만한 사람이 있나요?

결과 확인

- 5개 모두 체크: 훌륭합니다! 사업가의 마인드가 완성되었습니다.
 - 3~4개 체크: 잘하고 있습니다. 고객 후보를 더 생각해 보세요.
 - 1~2개 체크: 괜찮습니다. 수익 모델 캔버스부터 다시 채워 보세요.
 - 0개 체크: 걱정 마세요. Chapter 11을 다시 읽으면서 시작해 보세요.
-

당신도 할 수 있습니다

혹시 이런 생각이 드시나요?

“내 서비스에 돈을 낼 사람이 있을까?”
“수익화는 너무 어려운 것 아닌가?”
“사업이라니, 나한테는 너무 큰 이야기 같은데…”

한 가지 이야기를 해드리겠습니다.

Marc Lou의 첫 100달러

Marc Lou(마크 루)는 여러 제품을 만들었지만,
처음에는 아무도 돈을 내지 않았습니다.

몇 달을 무료로 운영했습니다.
포기하고 싶은 순간도 많았습니다.

그러다 어느 날, 첫 유료 고객이 생겼습니다.
100달러.

그는 그 순간을 이렇게 말했습니다.

“첫 100달러는 100만 달러보다 소중하다.
누군가 내가 만든 것에 돈을 냈다는 것은
이것이 진짜 가치가 있다는 증거다.”

여러분의 첫 수익도 그렇게 시작됩니다

첫 고객 한 명만 찾으면 됩니다.
그 한 명이 여러분에게 확신을 줄 것입니다.
“내가 만든 것에 가치가 있구나.”

이 확신이 있으면 나머지는 따라옵니다.
천천히, 하지만 확실하게 성장할 수 있습니다.

Part 5 미리보기: 지속 가능한 성장

Part 4에서는 수익화 계획을 세웠습니다.
Part 5에서는 [오래 지속할 수 있는 사업](#)을 만들어 봅니다.

- Chapter 15: 반복 작업을 자동화하기
- Chapter 16: 나만의 커뮤니티 만들기
- Chapter 17: 장기 비전 세우기

Part 5를 마치면 [1인 사업가로서의 로드맵](#)이 완성됩니다.
단순한 부업이 아닌, 지속 가능한 사업의 기초를 다져 봅시다.

준비되셨나요?
다음 장을 넘겨 주세요.

Chapter 15. 자동화로 시간 벌기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. 반복되는 작업을 **자동화**하는 방법을 알게 됩니다.
 2. **AI 챗봇**으로 고객 응대를 자동으로 처리하는 방법을 이해합니다.
 3. **모니터링** 자동 설정으로 문제를 미리 발견하는 방법을 배웁니다.
-

1. 반복 작업 자동화하기

매일 같은 일을 반복하고 계신가요?

서비스를 운영하다 보면 같은 일이 반복됩니다.

- 매일 아침 매출 확인하기
- 새 가입자에게 환영 이메일 보내기
- 매주 사용자 통계 정리하기
- 문의 메일에 비슷한 답변 보내기

이런 일들을 매번 직접 하시나요?

그러면 하루가 모자랍니다.

반복되는 일은 기계에게 맡기세요.

이메일 자동 응답 설정하기

가장 쉬운 자동화부터 시작합시다.

이메일 자동 응답입니다.

AI에게 이렇게 말하면 됩니다.

“새 사용자가 가입하면 환영 이메일을 자동으로 보내줘.
이메일에는 시작 가이드 링크를 포함해 줘.
3일 뒤에 사용법 안내 이메일도 보내줘.”

이렇게 한 번 설정하면 됩니다.

그다음부터는 자동으로 처리됩니다.

새벽에 가입한 사용자에게도 이메일이 갑니다.
내가 자고 있는 동안에도요.

정기 업데이트 자동화

콘텐츠를 정기적으로 업데이트해야 한다면?

이것도 자동화할 수 있습니다.

예를 들어 볼까요?

- [날씨 정보 사이트](#): 매시간 자동으로 데이터 갱신
- [가격 비교 서비스](#): 매일 새 가격 수집
- [뉴스 큐레이션](#): 매일 아침 관련 뉴스 자동 수집

AI에게 "매일 아침 9시에 이 작업을 해줘"라고 설정합니다.

이런 걸 [스케줄링](#)이라고 합니다.

정해진 시간에 자동으로 실행되는 것입니다.

2. AI 챗봇으로 고객 응대하기

같은 질문이 계속 옵니다

서비스를 운영하면 고객 문의가 옵니다.

신기한 건 대부분 비슷한 질문이라는 겁니다.

“비밀번호를 잊어버렸어요.”
“결제는 어떻게 하나요?”
“환불이 가능한가요?”
“이 기능은 어떻게 쓰나요?”

전체 문의의 80%가 같은 질문입니다.
이걸 매번 직접 답하면 시간 낭비입니다.

AI 챗봇을 세워 두세요

AI 챗봇이 대신 답변해 줍니다.

AI에게 이렇게 설정하면 됩니다.

“내 서비스에 고객 응대 챗봇을 추가해 줘.
자주 묻는 질문과 답변은 이거야. (목록 제공)
챗봇이 모르는 질문은 내 이메일로 전달해 줘.”

이렇게 하면 이런 흐름이 만들어집니다.

1. 고객이 질문합니다
2. AI 챗봇이 답변을 시도합니다
3. 답변할 수 있으면 바로 해결됩니다
4. 못 하면 나에게 알려줍니다

전체 문의의 80%가 자동으로 처리됩니다.

나는 나머지 20%만 신경 쓰면 됩니다.

챗봇이 점점 똑똑해집니다

AI 챗봇의 좋은 점이 또 있습니다.
새로운 질문이 오면 답변을 추가할 수 있습니다.

처음엔 20개 질문에 답할 수 있었습니다.

한 달 뒤엔 50개, 석 달 뒤엔 100개가 됩니다.

시간이 지날수록 챗봇이 더 잘 답합니다.

내가 직접 답해야 할 일은 점점 줄어듭니다.

3. 모니터링 자동 설정

문제가 생기면 바로 알아야 합니다

서비스가 갑자기 멈출 수 있습니다.

결제 시스템에 오류가 날 수도 있습니다.

사용자가 급격히 줄어들 수도 있습니다.

이런 문제를 모르고 있으면 큰일입니다.

자동으로 감시해 주는 시스템이 필요합니다.

어떤 것들을 모니터링할까요?

AI에게 이렇게 부탁하세요.

“내 서비스가 정상 작동하는지 자동으로 확인해 줘.

문제가 생기면 내 카카오톡으로 알림 보내줘.”

모니터링할 항목들입니다.

- **서비스 상태:** 사이트가 열리는지
- **에러 발생:** 오류가 나고 있는지
- **결제 상태:** 결제가 잘 되는지
- **사용자 수:** 갑자기 줄어들지 않는지
- **서버 용량:** 저장 공간이 부족하지 않은지

이걸 매번 직접 확인할 필요가 없습니다.

문제가 생겼을 때만 알림이 옵니다.

자동 백업도 필수입니다

데이터가 날아가면 끝입니다.

자동 백업을 반드시 설정하세요.

“매일 밤 12시에 데이터를 자동으로 백업해 줘.
최근 30일간의 백업을 보관해 줘.”

이 한 번의 설정이 큰 사고를 막아줍니다.

실생활 비유: 세탁기와 식기세척기

자동화를 생활 가전에 비유해 보겠습니다.

옛날에는 다 손으로 했습니다

할머니 세대를 생각해 보세요.

빨래를 하려면 빨래판에 비벼야 했습니다.

헹구고, 짜고, 널어야 했습니다.

빨래 하나에 한 시간이 걸렸습니다.

설거지도 마찬가지입니다.

하나하나 닦고, 헹구고, 물기를 닦았습니다.

하루의 대부분이 집안일이었습니다.

세탁기가 시간을 돌려줬습니다

세탁기가 생기고 나서 달라졌습니다.

빨래를 넣고 버튼만 누르면 됩니다.
세탁기가 돌아가는 동안 다른 일을 합니다.
식기세척기도 마찬가지입니다.
그릇을 넣고 버튼만 누르면 끝입니다.
돌려받은 시간으로 더 중요한 일을 합니다.

자동화는 디지털 세탁기입니다

서비스 운영도 똑같습니다.
이메일 보내기, 데이터 백업, 고객 응대.
이런 일들을 매번 손으로 할 필요가 없습니다.

자동화를 설정하면 버튼 하나로 끝납니다.
아니, 버튼도 필요 없습니다.
알아서 돌아갑니다.

내 시간을 돌려받는 것입니다.

절약한 시간으로 뭘 할 수 있을까요?
새로운 제품을 만들 수 있습니다.
가족과 시간을 보낼 수 있습니다.
취미를 즐길 수 있습니다.

자동화의 진짜 가치는 여기에 있습니다.

실제 사례: 1인 운영자의 하루

Pieter Levels는 하루 몇 시간 일할까요?

놀라운 사실이 있습니다.
연 38억을 버는 Pieter Levels는 하루 2~3시간만 일합니다.

나머지 시간은 뭘 할까요?
여행을 하거나 취미를 즐깁니다.

어떻게 이게 가능할까요?
답은 자동화입니다.

모든 것이 자동으로 돌아갑니다

그의 서비스는 이렇게 운영됩니다.

- **서버 모니터링**: 자동으로 감시, 문제 시 알림
- **결제 처리**: 구독 갱신, 환불 모두 자동
- **고객 FAQ**: 챗봇이 대부분의 질문에 답변
- **데이터 업데이트**: 정해진 시간에 자동 갱신
- **이메일 발송**: 가입, 결제, 알림 모두 자동

사람이 직접 해야 하는 일이 거의 없습니다.

“일하지 않아도 돈이 들어오는 구조”

Pieter Levels는 이렇게 말했습니다.

“내가 해변에서 쉬는 동안에도
누군가 내 서비스에 가입하고 결제한다.
이게 자동화의 힘이다.”

물론 처음에는 시간이 걸립니다.
자동화를 설정하는 데 노력이 필요합니다.
하지만 한 번 설정하면 계속 작동합니다.
처음 한 번의 수고가 평생의 편리함이 됩니다.
우리도 같은 구조를 만들 수 있습니다.
바이브 코딩이면 충분합니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 반복되는 일은 자동화 후보입니다. 같은 일을 세 번 이상 하고 있다면 자동화하세요.
 2. AI 챗봇으로 고객 질문의 80%를 자동 응대할 수 있습니다. 나는 나머지 20%만 처리합니다.
 3. 자동 백업과 모니터링은 필수입니다. 한 번 설정하면 큰 사고를 막아줍니다.
 4. 절약한 시간으로 새로운 것을 만드세요. 자동화의 진짜 가치는 돌려받은 시간입니다.
 5. 처음 설정할 때 시간이 들지만, 한 번 만들면 계속 작동합니다.
-

다음 챕터 미리보기

Chapter 16에서는 나만의 커뮤니티 만들기를 배웁니다.

- 사용자 커뮤니티가 왜 중요한지
- 카카오톡 오픈채팅으로 시작하는 방법
- 커뮤니티가 제품을 키워주는 원리

혼자 달리는 것보다 함께 가는 게 더 멀리 갑니다.

커뮤니티의 힘을 알아보겠습니다.

Chapter 16. 나만의 커뮤니티 만들기

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. 사용자 커뮤니티가 사업에 왜 중요한지 이해합니다.
 2. 카카오톡 오픈채팅이나 디스코드로 커뮤니티를 시작하는 방법을 알게 됩니다.
 3. 커뮤니티를 건강하게 운영하는 노하우를 배웁니다.
-

1. 사용자 커뮤니티의 힘

충성 고객은 어디서 올까요?

광고로 온 고객은 쉽게 떠납니다.

더 짠 곳이 나타나면 바로 옮겨갑니다.

하지만 커뮤니티 안의 고객은 다릅니다.

쉽게 떠나지 않습니다.

왜 그럴까요?

사람들은 제품이 아니라 사람에게 붙습니다.

커뮤니티에서 친구가 생깁니다.

도움을 주고받습니다.

소속감이 생깁니다.

이 소속감이 바로 충성의 비밀입니다.

커뮤니티가 주는 세 가지 선물

첫째, 고객이 알아서 홍보해 줍니다.

만족한 사용자가 친구에게 추천합니다.

“이 모임 좋으니까 너도 들어와.”

돈 한 푼 안 들이는 최고의 마케팅입니다.

둘째, 제품 아이디어가 쏟아집니다.

사용자들이 이야기를 나눕니다.

“이런 기능 있으면 좋겠다.”

“여기가 좀 불편하다.”

무료 시장 조사가 매일 이루어집니다.

셋째, 고객이 고객을 도와줍니다.

새로 온 사람이 질문을 합니다.

기존 회원이 답변해 줍니다.

내가 일일이 답하지 않아도 됩니다.

커뮤니티는 자동 고객 센터입니다.

2. 커뮤니티 시작하기

어디서 시작할까요?

거창한 플랫폼이 필요 없습니다.

익숙한 곳에서 시작하세요.

카카오톡 오픈채팅이 가장 쉽습니다.

- 모든 한국인이 카카오톡을 쓰고 있습니다
- 앱을 새로 설치할 필요가 없습니다
- 오픈채팅방 만드는 데 1분이면 됩니다

- 무료입니다

조금 더 전문적으로 하고 싶다면 [디스코드](#)도 좋습니다.
디스코드는 주제별 채널을 나눌 수 있습니다.
게임 커뮤니티에서 많이 쓰는 무료 채팅 앱입니다.

카카오톡 오픈채팅으로 시작하는 법

정말 간단합니다.

1. 카카오톡을 엽니다
2. 채팅 탭에서 "오픈채팅"을 선택합니다
3. "오픈채팅방 만들기"를 누릅니다
4. 방 이름과 소개를 적습니다
5. 완성입니다

방 이름은 이렇게 지으면 좋습니다.

“[서비스 이름] 사용자 모임”
“[서비스 이름] 함께 성장하기”

처음에 몇 명이면 될까요?

처음부터 많을 필요 없습니다.

[5명이면 충분합니다.](#)

5명이 활발하게 이야기하면 분위기가 만들어집니다.
그 분위기를 보고 새 사람이 들어옵니다.
자연스럽게 늘어납니다.

처음 5명은 어디서 찾을까요?

- 내 서비스의 초기 사용자
- 주변 지인 중 관심 있는 사람
- SNS에서 비슷한 관심사를 가진 사람

“우리 서비스 사용자 모임을 만들었어요.
함께 이야기 나눠요.”

이 한마디면 됩니다.

| 3. 커뮤니티 운영 노하우

규칙은 간단하게

규칙이 너무 많으면 사람들이 부담을 느낍니다.
3가지면 충분합니다.

1. 서로 존중하기 — 비난, 욕설 금지
2. 광고 금지 — 관련 없는 홍보 금지
3. 질문 환영 — 초보 질문도 환영합니다

이 세 가지만 지키면 건강한 분위기가 됩니다.

모임장이 너무 열심히 하면 안 됩니다

의외의 조언입니다.

모임장이 모든 질문에 답하면 안 됩니다.
모든 대화를 이끌면 안 됩니다.

왜냐하면 그러면 **나 없이는 안 되는 모임**이 됩니다.
내가 바쁘면 모임이 죽습니다.

대신 이렇게 하세요.

- 질문이 오면 다른 회원에게 넘기기
 - “이건 ○○님이 잘 아실 것 같아요”
 - 회원들끼리 대화할 거리를 던지기
 - “이번 주에 새로 시도한 것 공유해 주세요”

나는 분위기를 만드는 역할만 합니다.

실제 대화는 회원들이 합니다.

정기 이벤트를 만드세요

커뮤니티가 활발해지려면 이벤트가 필요합니다.

거창할 필요 없습니다.

- **매주 월요일**: “이번 주 목표 공유하기”
- **매주 금요일**: “이번 주 성과 자랑하기”
- **매월 첫째 주**: “온라인 모임 (줌/구글밋)”

정기적인 리듬이 생기면 사람들이 습관처럼 옵니다.

실생활 비유: 동네 모임

커뮤니티를 동네 모임에 비유해 보겠습니다.

등산 모임을 생각해 보세요

동네에 등산 모임이 있습니다.

매주 토요일 아침에 뒷산에 갑니다.

처음엔 3~4명이 모였습니다.

산을 오르며 이야기를 나눴습니다.

한 달 뒤에는 10명이 되었습니다.

두 달 뒤에는 20명이 되었습니다.

모임장이 특별한 걸 한 게 아닙니다.

시간과 장소만 정해 준 겁니다.

사람들이 알아서 친해졌습니다.

모임이 커지면 좋은 일이 생깁니다

등산 모임 회원 중에 의사가 있었습니다.

“건강 상담 무료로 해드릴게요.”

요리사도 있었습니다.

“등산 후 제가 국수 끓여 드릴게요.”

사진작가도 있었습니다.

“단체 사진 제가 찍어 드릴게요.”

사람이 모이면 예상 못한 가치가 생깁니다.

디지털 커뮤니티도 똑같습니다.

사용자 중에 전문가가 있습니다.

서로 돋다 보면 커뮤니티 자체가 가치가 됩니다.

모임 자체가 이유가 됩니다

처음엔 등산 때문에 모였습니다.

나중엔 사람들 때문에 옵니다.

“산은 안 가도 모임엔 나가고 싶어.”

커뮤니티도 마찬가지입니다.

처음엔 서비스 때문에 들어왔습니다.

나중엔 커뮤니티 자체가 목적이 됩니다.

이것이 진짜 충성 고객입니다.

실제 사례: NomadList 커뮤니티

정보 사이트에서 커뮤니티로 진화

NomadList는 처음에 정보 사이트였습니다.

도시별 물가, 인터넷 속도, 날씨 정보를 제공했습니다.

그런데 사용자들이 모이기 시작했습니다.
"이 도시 어때요?"라고 서로 물었습니다.
"같이 만날 사람?"이라고 약속을 잡았습니다.

Pieter Levels는 이 기회를 놓치지 않았습니다.
슬랙 채널을 열었습니다.
슬랙은 업무용 메신저 앱입니다.
카카오톡 단체방과 비슷하다고 생각하시면 됩니다.

슬랙 채널이 수백 개가 되었습니다

도시별 채널이 생겼습니다.
서울 채널, 방콕 채널, 리스본 채널.
주제별 채널도 생겼습니다.
프리랜서 채널, 건강 채널, 취미 채널.

수백 개의 채널에서 회원들이 대화합니다.
Pieter Levels가 일일이 관리하지 않습니다.
회원들이 알아서 운영합니다.

커뮤니티가 제품의 핵심 가치가 되었습니다

재미있는 일이 벌어졌습니다.
도시 정보는 다른 사이트에서도 볼 수 있습니다.
하지만 NomadList 커뮤니티는 여기밖에 없습니다.
사람들이 NomadList에 돈을 내는 이유가 바뀌었습니다.

처음: "도시 정보가 유용해서"
지금: "커뮤니티에 속하고 싶어서"

커뮤니티 자체가 제품이 된 것입니다.
이것이 커뮤니티의 진짜 힘입니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. 커뮤니티가 제품을 키워줍니다. 충성 고객, 무료 마케팅, 제품 아이디어까지 얻을 수 있습니다.
 2. 카카오톡 오픈채팅으로 오늘 바로 시작할 수 있습니다. 5명이면 충분합니다.
 3. 서로 돋는 문화를 만드세요. 모임장이 다 하지 말고 회원들이 주인공이 되게 하세요.
 4. 커뮤니티 자체가 수익원이 될 수 있습니다. NomadList처럼 커뮤니티가 제품의 핵심 가치가 됩니다.
 5. 정기 이벤트로 리듬을 만들면 사람들이 습관처럼 찾아옵니다.
-

다음 챕터 미리보기

Chapter 17에서는 1인 사업가의 장기 비전을 그려봅니다.

- 벤아웃 없이 지속하는 방법
- 건강과 균형의 중요성
- AI 기술의 미래와 새로운 기회

오래 달리려면 자기 페이스를 알아야 합니다.

마지막 챕터에서 함께 이야기해 보겠습니다.

Chapter 17. 1인 사업가의 장기 비전

학습 목표

이 챕터를 읽고 나면 이런 것들을 알게 됩니다.

1. **변아웃 없이** 사업을 지속하는 방법을 알게 됩니다.
 2. **건강과 균형**이 사업의 기반임을 이해합니다.
 3. **AI 기술의 미래**와 앞으로의 기회를 그려볼 수 있습니다.
-

1. 지속 가능한 사업

변아웃은 누구에게나 옵니다

처음 시작할 때는 열정이 넘칩니다.

밤늦게까지 작업합니다.

주말에도 일합니다.

한 달쯤 지나면 슬슬 지칩니다.

두 달이면 “이걸 왜 시작했지?” 생각이 듭니다.

이것이 **변아웃**입니다.

몸과 마음이 지쳐서 더 이상 못 하겠는 상태입니다.

변아웃은 나약해서 오는 게 아닙니다.

너무 열심히 해서 오는 겁니다.

지속 가능한 1인 사업의 비결

단거리 질주



마라톤



- 매일 1시간 > 주말 10시간



- 작은 성공 축적 → 큰 결과



- 완벽보다 완성, 완성보다 지속

페이스 조절이 답입니다

1인 사업은 단거리 달리기가 아닙니다.

마라톤입니다.

마라톤 선수가 처음부터 전력 질주하면 어떻게 될까요?

5km도 못 가서 쓰러집니다.

사업도 마찬가지입니다.

하루 2~3시간이면 충분합니다.

매일 꾸준히 하는 게 중요합니다.

하루 12시간 x 1개월보다

하루 2시간 x 12개월이 훨씬 좋습니다.

쉬는 것도 일입니다

죄책감 없이 쉬세요.

쉬는 동안 뇌가 정리를 합니다.

산책을 하다가 아이디어가 떠오릅니다.

잠을 자다가 해결책이 생각납니다.

쉬는 시간은 낭비가 아닙니다.

오히려 생산성을 높이는 시간입니다.

일주일에 최소 하루는 완전히 쉬세요.

그날은 컴퓨터를 열지 마세요.

서비스 알림도 꺼 두세요.

2. 건강과 균형

몸이 자본입니다

1인 사업의 가장 큰 리스크가 뭔지 아시나요?

경쟁자가 아닙니다.

기술 변화도 아닙니다.

내 건강입니다.

직원이 없는 1인 사업은 내가 쓰러지면 끝입니다.

아무도 대신해 줄 사람이 없습니다.

그래서 건강이 최우선입니다.

간단한 건강 루틴

거창할 필요 없습니다.

아침: 30분 산책하기

컴퓨터 앞에 앉기 전에 밖에 나가세요.

햇빛을 쬐면 하루가 달라집니다.

작업 중: 50분 일하고 10분 쉬기

타이머를 맞춰 놓으세요.

10분 동안 스트레칭을 하세요.

저녁: 정해진 시간에 컴퓨터 끄기

"오늘은 여기까지"라고 선을 그으세요.

저녁 시간은 가족과 나를 위한 시간입니다.

가족과의 균형

50대, 60대에 사업을 시작하면 좋은 점이 있습니다.

인생 경험이 풍부합니다.

무엇이 진짜 중요한지 알고 있습니다.

돈을 많이 벌어도 가족이 불행하면 의미가 없습니다.

사업은 인생을 풍요롭게 하는 도구일 뿐입니다.

가족에게 내 사업 이야기를 들려주세요.

응원을 받으면 힘이 납니다.

때로는 가족이 최고의 조언자가 됩니다.

3. AI 기술의 미래와 기회

AI는 계속 발전합니다

지금도 놀라운데, 앞으로는 더 대단해집니다.

2025년의 AI와 2030년의 AI는 크게 다를 겁니다.

마치 2015년 스마트폰과 2025년 스마트폰이 다르듯이요.

이것은 위기가 아닙니다. 기회입니다.

AI가 발전할수록 1인 사업가가 할 수 있는 일이 늘어납니다.
지금은 간단한 웹사이트를 만들 수 있습니다.
앞으로는 더 복잡한 서비스도 혼자 만들 수 있습니다.

앞으로 더 쉬워지는 것들

AI 도구가 더 똑똑해집니다.

지금은 구체적으로 설명해야 합니다.
앞으로는 대충 말해도 알아듣습니다.

비용이 더 낮아집니다.

기술이 발전하면 가격이 내려갑니다.
더 적은 비용으로 더 많은 일을 할 수 있습니다.

새로운 분야가 열립니다.

AI 영상 제작, AI 음악, AI 교육.
지금은 상상도 못 하는 사업이 가능해집니다.

지금 시작하는 사람이 유리합니다

AI 시대는 이제 시작입니다.
지금 바이브 코딩을 배우고 있는 여러분은 [선구자](#)입니다.

남들이 "AI가 뭐야?"라고 할 때
여러분은 이미 AI로 제품을 만들고 있습니다.

남들이 "나도 해볼까?"라고 할 때
여러분은 이미 수익을 내고 있습니다.

이 경험의 차이는 시간이 지날수록 벌어집니다.

실생활 비유: 마라톤

장기 비전을 마라톤에 비유해 보겠습니다.

전력 질주하면 쓰러집니다

마라톤 초보가 흔히 하는 실수가 있습니다.

출발선에서 전력 질주하는 것입니다.

주변 사람들이 빠르게 달리니까 따라갑니다.

1km까지는 신이 납니다.

3km에서 숨이 차옵니다.

5km에서 포기하고 싶어집니다.

사업도 똑같습니다.

남들이 빨리 성공하는 것 같아 보입니다.

조급해집니다.

무리하게 됩니다.

자기 페이스로 달리세요

경험 많은 마라톤 선수는 다릅니다.

자기 페이스를 지킵니다.

처음 10km는 천천히 갑니다.

중반에 속도를 조금 올립니다.

마지막에 남은 힘으로 마무리합니다.

완주가 목표입니다.

1등이 아니어도 됩니다.

사업도 마찬가지입니다.

남과 비교하지 마세요.

내 속도로 꾸준히 가면 됩니다.

중간에 걸어도 괜찮습니다

마라톤에서 걷는 구간이 있어도 됩니다.

잠깐 쉬어도 됩니다.

물 한 잔 마시고 다시 가면 됩니다.

사업도 마찬가지입니다.

잘 안 되는 시기가 옵니다.

쉬고 싶을 때가 있습니다.

그때 멈추는 게 아니라 걷는 겁니다.

속도를 줄여도 됩니다.

쉬었다 다시 해도 됩니다.

포기하지 않는 것이 중요합니다.

결승선은 반드시 있습니다.

실제 사례: Pieter Levels 10년의 여정

2014년, 바닥이었습니다

Pieter Levels의 시작은 화려하지 않았습니다.

2014년, 그는 이런 상태였습니다.

- 부모님 집에 얹혀살고 있었습니다
- 우울증을 겪고 있었습니다
- 미래가 보이지 않았습니다
- 할 수 있는 건 컴퓨터 앞에 앉는 것뿐이었습니다

그가 선택한 건 **작게 시작하기**였습니다.

대단한 계획이 아니었습니다.

“일단 뭐라도 만들어 보자.”

10년간 꾸준히

2014년부터 2024년까지.

10년이라는 시간입니다.

그 사이에 많은 일이 있었습니다.

- 실패한 프로젝트가 수십 개 있었습니다
- 벤아웃도 여러 번 겪었습니다
- 사람들의 비판도 받았습니다
- 포기하고 싶은 순간도 많았습니다

하지만 그는 한 가지를 지켰습니다.

“작지만 매일 조금씩”

어떤 날은 1시간만 일했습니다.

어떤 날은 산책만 하고 돌아왔습니다.

하지만 완전히 멈추지는 않았습니다.

벤아웃을 이긴 방법

Pieter Levels도 벤아웃을 겪었습니다.

그가 극복한 방법은 이것이었습니다.

페이스 조절.

바쁠 때는 하루 8시간도 일했습니다.

지칠 때는 하루 1시간만 일했습니다.

때로는 일주일간 여행을 떠났습니다.

“멈추지 않되 무리하지 않는다.”

이것이 10년을 버틸 수 있었던 비결입니다.

2024년, 연 38억

10년이 지난 지금 결과를 보겠습니다.

- 연 수입 약 38억 원
- 직원 0명, 혼자 운영
- 세계 어디서든 일할 수 있는 자유
- 하루 2~3시간만 일하는 생활

하루아침에 이루어진 게 아닙니다.
10년간 조금씩 쌓아 올린 결과입니다.
여러분에게도 이 시간이 있습니다.

핵심 포인트

이 챕터에서 꼭 기억할 것들입니다.

1. **작지만 꾸준한 성장**이 최고의 전략입니다. 매일 조금씩, 멈추지 마세요.
 2. **건강과 균형**이 사업의 기반입니다. 몸이 무너지면 사업도 무너집니다.
 3. **AI 기술은 계속 발전**합니다. 지금 시작하면 앞으로 더 많은 기회가 열립니다.
 4. **지금 시작한 당신이 선구자입니다.** 남들보다 먼저 시작한 경험은 큰 자산이 됩니다.
 5. **마라톤처럼 자기 페이스로 달리세요.** 중간에 걸어도 괜찮습니다. 포기만 안 하면 됩니다.
-

마무리

여기까지 오신 여러분께

Chapter 1에서 "바이브 코딩이 뭐지?"라고 시작하셨습니다.
코딩이라는 단어가 낯설고 두려웠을 수도 있습니다.

그런데 지금은 어떠신가요?

AI 도구가 무엇인지 알게 되었습니다.
아이디어를 제품으로 만드는 법을 배웠습니다.
수익을 내는 방법도 알게 되었습니다.
자동화로 시간을 절약하는 법도 배웠습니다.
커뮤니티를 만드는 법도 알게 되었습니다.

이 모든 도구가 이제 여러분의 것입니다.

50대, 60대라서 늙은 게 아닙니다

오히려 유리한 점이 많습니다.

수십 년간 쌓아온 인생 경험이 있습니다.

사람들이 진짜 필요한 것이 무엇인지 알고 있습니다.

인내심과 끈기가 있습니다.

급하게 서두르지 않는 지혜가 있습니다.

젊은 사람들은 기술은 빠르지만

여러분만의 깊은 통찰은 없습니다.

기술은 AI가 채워줍니다.

통찰은 여러분만 가지고 있습니다.

작게, 하지만 꾸준히

이 책에서 가장 드리고 싶은 말씀은 이것입니다.

완벽하지 않아도 시작하세요.

작아도 괜찮으니 매일 조금씩 하세요.

Pieter Levels도 부모님 집에서 시작했습니다.

여러분도 지금 있는 자리에서 시작하면 됩니다.

컴퓨터 하나와 AI 도구만 있으면 됩니다.

나머지는 여러분의 경험과 아이디어가 채워줄 겁니다.

여러분의 사업을 응원합니다

이 책이 작은 디딤돌이 되었으면 합니다.

첫 발을 내딛는 데 용기를 드렸다면 더 바랄 게 없습니다.

Part 5 실습이 마지막으로 남아 있습니다.

지금까지 배운 모든 것을 종합해서

여러분만의 장기 사업 계획을 세워 보겠습니다.

새로운 시작을 축하드립니다.

여러분의 바이브 코딩 여정을 진심으로 응원합니다.

화이팅!

Part 5 마무리: 지속 가능한 1인 사업 로드맵

Part 5 핵심 요약

세 개의 챕터를 통해 이런 것들을 배웠습니다.

Chapter 15 — 반복 작업을 자동화하기

- 매일 반복하는 일을 AI에게 맡길 수 있습니다.
- 고객 응대, 데이터 정리, 알림 발송 등을 자동화할 수 있습니다.
- 자동화는 거창한 것이 아닙니다. **하나씩** 줄여 나가면 됩니다.

Chapter 16 — 나만의 커뮤니티 만들기

- 혼자 하면 외롭지만, 함께하면 **오래** 갈 수 있습니다.
- 카카오톡 오픈채팅, 네이버 카페 등으로 시작할 수 있습니다.
- 커뮤니티는 **마케팅 채널**이자 **응원단**이 됩니다.

Chapter 17 — 장기 비전 세우기

- 1인 사업은 **마라톤**입니다. 단거리 달리기가 아닙니다.
 - 번아웃을 예방하는 것도 중요한 전략입니다.
 - 1년, 3년, 5년 후를 **상상해** 보는 것이 비전의 시작입니다.
-

실습 과제: 나의 12개월 로드맵 만들기

계획이 있으면 흔들리지 않습니다.

앞으로 1년간 무엇을 할지 정리해 봅시다.

나의 12개월 로드맵

각 기간에 해야 할 일을 적어 보세요.

완벽하지 않아도 됩니다. 나중에 얼마든지 수정할 수 있습니다.

1~3개월: MVP 완성하기

이 기간의 목표: **최소한의 기능을 갖춘 첫 버전 완성**

만들 제품/서비스: _____

핵심 기능 (최대 3개):

1. _____

2. _____

3. _____

완성 목표일: _____ 월 _____ 일

4~6개월: 첫 수익 만들기

이 기간의 목표: **첫 유료 고객 확보**

수익 모델: _____

목표 가격: _____

첫 고객 후보 3명:

1.

2.

3.

월 수익 목표: _____ 원

7~9개월: 마케팅과 성장

이 기간의 목표: 고객을 늘리고 서비스 개선

마케팅 채널:

1.

2.

자동화할 작업:

1.

목표 고객 수: _____ 명

월 수익 목표: _____ 원

10~12개월: 확장과 안정화

이 기간의 목표: 안정적인 수익 구조 만들기

추가할

기능이나

서비스:

커뮤니티 운영 계획: _____

새로운

수익원

(있다면): _____

월 수익 목표: _____ 원

1년 후 나의 모습: _____

로드맵 예시

기간	목표	구체적 할 일
1~3개월	MVP 완성	“오늘의 텃밭 일기” 기본 기능 완성, 지인 10명 테스트
4~6개월	첫 수익	프리미엄 기능 추가, 유료 전환 목표 20명
7~9개월	마케팅	네이버 카페 홍보, 블로그 글 주 1회 작성
10~12개월	확장	월 50만 원 수익 안정화, 두 번째 서비스 기획

이 로드맵은 약속이 아닙니다.

방향을 잡기 위한 나침반입니다.

상황에 따라 자유롭게 바꾸세요.

자가 진단 체크리스트

Part 5를 잘 마쳤는지 스스로 점검해 보세요.

아래 항목을 읽고 체크해 보세요.

- [] 자동화할 수 있는 작업을 1개 이상 찾았다.
→ 매일 반복하는 일 중에 AI가 대신할 수 있는 것이 있나요?
- [] 커뮤니티를 시작할 방법을 정했다.
→ 오픈채팅, 카페, 디스코드 등 하나를 골랐나요?

- [] 12개월 목표를 세웠다.
→ 위 로드맵 빙칸을 채웠다면 성공입니다
- [] 번아웃 예방 계획이 있다.
→ 쉬는 날, 취미 시간 등을 정해 두었나요?
- [] "나도 할 수 있다"는 확신이 생겼다.
→ 이 확신이 가장 중요합니다!

결과 확인

- 5개 모두 체크: 축하합니다! 이미 1인 사업가의 마인드를 갖추셨습니다.
 - 3~4개 체크: 아주 잘하고 있습니다. 나머지는 실천하면서 채우면 됩니다.
 - 1~2개 체크: 괜찮습니다. 12개월 로드맵부터 다시 작성해 보세요.
 - 0개 체크: 걱정 마세요. Chapter 15부터 천천히 다시 읽어 보세요.
-

당신도 할 수 있습니다

이 책의 마지막 실습입니다.

여기까지 오신 것만으로도 대단합니다.

혹시 이런 생각이 드시나요?

“정말 이대로 1인 사업을 할 수 있을까?”

“아직 준비가 부족한 것 같은데...”

“나는 특별한 사람이 아닌데...”

마지막 이야기를 해드리겠습니다.

Pieter Levels의 10년 여정

Pieter Levels는 하루아침에 성공한 것이 아닙니다.

- 2014년에 첫 프로젝트를 시작했습니다.
- 수많은 실패를 겪었습니다.
- 몇 년 동안 수익이 거의 없었습니다.

10년이 걸렸습니다.

하지만 그는 멈추지 않았습니다.

작은 것을 하나씩, 꾸준히 만들어 갔습니다.

그리고 지금은 연 38억 원을 벌고 있습니다.
직원 없이, 혼자서.

여러분의 시작이 가장 용감한 한 걸음입니다

Pieter Levels에게 10년이 있었다면,
여러분에게도 시간이 있습니다.

다른 점이 있다면,
여러분에게는 AI가 있습니다.

코딩을 배우지 않아도 됩니다.
디자인을 공부하지 않아도 됩니다.
아이디어만 있으면 됩니다.

그리고 여러분에게는 아이디어가 있습니다.
수십 년의 경험에서 나온 진짜 아이디어가 있습니다.

**이 책을 펼친 순간,
여러분은 이미 가장 용감한 한 걸음을 내디뎠습니다.**

그 한 걸음이 1인 사업가의 시작입니다.
천천히, 하지만 멈추지 말고 걸어가세요.
여러분을 응원합니다.

에필로그 안내

이 책의 모든 실습을 마쳤습니다.

정말 수고 많으셨습니다.

에필로그에서는 이 책의 여정을 마무리합니다.

- 지금까지 배운 것을 [한눈에](#) 정리합니다.
- 앞으로 도움이 될 [추가 자료](#)를 안내합니다.
- 1인 사업가 커뮤니티에 [참여하는 방법](#)을 알려드립니다.
- 저자의 마지막 인사가 있습니다.

마지막 페이지를 넘겨 주세요.

여러분의 새로운 시작을 축하합니다.

에필로그: 당신의 이야기는 이제 시작입니다

여기까지 오신 당신에게

이 책을 끝까지 읽어주셔서 감사합니다.

프롤로그에서 두 사람의 이야기를 들려드렸습니다.

부모님 집에서 시작한 Pieter Levels.

27번 실패한 Marc Lou.

그들의 이야기가 특별해 보였을 수 있습니다.

하지만 지금 이 책을 다 읽은 여러분은 아실 겁니다.

그들이 한 일은 특별하지 않았습니다.

아이디어를 떠올리고.

도구를 사용해서 만들고.

세상에 공개하고.

피드백을 받아 개선하고.

수익을 만들고.

꾸준히 성장했습니다.

이 과정을 여러분도 이미 알게 되었습니다.

여러분이 얻은 것들

이 책을 통해 여러분은 이런 것들을 배웠습니다.

Part 1에서는 마음가짐을 다렸습니다.
바이브 코딩이 뭔지 알게 되었습니다.
코딩을 몰라도 된다는 걸 확인했습니다.

Part 2에서는 직접 해봤습니다.
AI 도구를 설치하고 첫 대화를 나눴습니다.
웹페이지를 만들어 세상에 공개했습니다.

Part 3에서는 실전 프로젝트를 경험했습니다.
기획하고, 만들고, 다듬고, 피드백을 받았습니다.

Part 4에서는 수익화를 배웠습니다.
결제 연동, 마케팅, 수익 확장 방법을 알게 되었습니다.

Part 5에서는 지속 가능한 성장을 설계했습니다.
자동화, 커뮤니티, 장기 비전까지 그려봤습니다.

이 모든 것을 해낸 여러분이 대단합니다.

한 가지 부탁

이 책을 덮기 전에 한 가지만 해주세요.

Part 1 실습에서 적었던 "만들고 싶은 것 3가지"를 다시 보세요.

처음에 적었을 때와 지금의 느낌이 다를 겁니다.
처음에는 "이걸 정말 할 수 있을까?" 했을 거예요.
지금은 "어떻게 만들면 되겠다"가 보일 겁니다.

그 차이가 이 책의 성과입니다.

완벽하지 않아도 시작하세요

마지막으로 한 가지만 더 말씀드리겠습니다.

Pieter Levels의 첫 사이트는 투박했습니다.

Marc Lou의 첫 제품은 아무도 쓰지 않았습니다.

하지만 그들은 시작했습니다.

완벽하지 않은 상태에서 시작했습니다.

완벽함은 시작의 적입니다.

지금 당장, 오늘, 작은 것 하나를 시작하세요.

- AI에게 "안녕"이라고 말해보세요
- 간단한 웹페이지를 만들어보세요
- 주변 한 사람에게 보여주세요

그 작은 한 걸음이 38억의 시작이 될 수 있습니다.

Pieter Levels가 당신에게 하고 싶은 말

Pieter Levels는 인터뷰에서 이렇게 말한 적이 있습니다.

“시작하기 전에 모든 걸 알 필요는 없다.

시작하면서 배우면 된다.

가장 중요한 건 시작하는 것이다.”

이 말을 여러분에게 전합니다.

여러분의 이야기는 이제 시작입니다.

이 책이 그 시작의 작은 불씨가 되었기를 바랍니다.

응원합니다.

언제나 응원하겠습니다.

감사합니다.

부록

A. 유용한 도구 목록

이 책에서 소개한 도구들을 정리했습니다.

북마크해 두고 필요할 때 찾아보세요.

AI 코딩 도구

도구	설명	가격	주소
Claude	AI 대화 도구, 코드 작성	무료/월 \$20	claude.ai
Cursor	AI 내장 코드 편집기	무료/월 \$20	cursor.com
Bolt	대화로 웹사이트 생성	무료/유료	bolt.new
Lovable	대화로 앱 생성	무료/유료	lovable.dev
v0	UI 디자인 AI 생성	무료/유료	v0.dev

배포 & 호스팅

도구	설명	가격	주소
Netlify	무료 웹 배포 서비스	무료/유료	netlify.com
Vercel	무료 웹 배포 서비스	무료/유료	vercel.com
GitHub Pages	무료 정적 사이트 호스팅	무료	pages.github.com

결제 시스템

도구	설명	주소
Stripe	글로벌 결제 시스템	stripe.com
토스페이먼츠	국내 결제 시스템	tosspayments.com
Gumroad	디지털 제품 판매	gumroad.com
Lemon Squeezy	디지털 제품 판매	lemonsqueezy.com

디자인 & 이미지

도구	설명	주소
Canva	온라인 디자인 도구	canva.com
Figma	UI 디자인 도구	figma.com
Unsplash	무료 고품질 사진	unsplash.com

마케팅 & 커뮤니티

도구	설명	주소
X (트위터)	Build in Public 공유	x.com
Product Hunt	제품 런칭 플랫폼	producthunt.com
카카오톡 오픈채팅	국내 커뮤니티 운영	open.kakao.com
디스코드	글로벌 커뮤니티 운영	discord.com

B. 자주 묻는 질문 (FAQ)

Q1. 정말 코딩을 하나도 몰라도 되나요?

네, 됩니다.

이 책에서 소개하는 바이브 코딩은 코드를 직접 작성하지 않습니다.

AI에게 한국어로 원하는 것을 설명하면 됩니다.

카카오톡 메시지 보내는 것처럼 대화하면 됩니다.

Q2. 비용이 얼마나 드나요?

무료로 시작할 수 있습니다.

- Claude: 무료 버전으로 시작
- Netlify: 무료 배포
- 도메인: 무료 주소 제공

유료로 전환하더라도 월 2~3만 원이면 충분합니다.

커피 한 잔 값으로 사업을 시작할 수 있습니다.

Q3. 영어를 못해도 되나요?

한국어로 충분합니다.

Claude는 한국어를 잘 이해합니다.

한국어로 설명하면 한국어로 답해줍니다.

도구 화면이 영어여도 AI에게 물어보면 됩니다.

Q4. 컴퓨터 사양이 좋아야 하나요?

일반 노트북이면 충분합니다.

인터넷 브라우저만 열 수 있으면 됩니다.

대부분의 작업은 웹에서 이루어집니다.

최신 고사양 컴퓨터는 필요 없습니다.

Q5. 수익이 정말 나나요?

사람마다 다릅니다.

솔직히 모든 사람이 38억을 벌지는 않습니다.

하지만 월 10만 원, 50만 원은 충분히 가능합니다.

중요한 건 시작하느냐 아니냐입니다.

시작하지 않으면 수익은 영원히 0입니다.

Q6. 혼자서 할 수 있나요?

네, 혼자서 할 수 있습니다.

Pieter Levels도 혼자 운영합니다.

Marc Lou도 혼자 운영합니다.

AI가 팀원 역할을 해주기 때문입니다.

물론 주변에 함께할 사람이 있으면 더 좋습니다.

하지만 혼자서도 충분히 가능합니다.

Q7. 실패하면 어떡하나요?

실패해도 괜찮습니다.

Marc Lou는 27번 실패했습니다.

실패는 끝이 아니라 배움입니다.

바이브 코딩의 장점은 실패 비용이 거의 없다는 것입니다.

돈이 거의 안 들고, 시간도 많이 안 걸립니다.

실패하면 다른 것을 만들면 됩니다.

Q8. 어디서 더 배울 수 있나요?

아래 커뮤니티와 채널을 추천합니다.

- **이 책의 독자 커뮤니티:** [오픈채팅 링크]
- **유튜브:** “바이브 코딩”, “vibe coding” 검색

- X (트위터): #BuildInPublic 해시태그
 - 인디해커스: indiehackers.com
-

C. 용어 사전

이 책에서 사용한 주요 용어들을 정리했습니다.

용어

뜻

AI (인공지능) 사람처럼 생각하고 대화할 수 있는 컴퓨터 프로그램

바이브 코딩 AI에게 원하는 것을 말로 설명해서 프로그램을 만드는 방식

프롬프트 AI에게 주는 지시나 요청 문장

웹사이트 인터넷에서 볼 수 있는 페이지 (네이버, 구글 같은 것)

웹 앱 인터넷 브라우저에서 사용하는 프로그램

배포 (Deploy) 만든 웹사이트를 인터넷에 올리는 것

도메인 인터넷 주소 (예: naver.com, google.com)

MVP 최소 기능 제품. 핵심 기능만 담은 첫 번째 버전

프리미엄 모델 기본은 무료, 추가 기능은 유료인 사업 방식

구독 모델 매달 일정 금액을 내고 서비스를 이용하는 방식

Build in Public 만드는 과정을 SNS에 공개하는 마케팅 방법

피드백 사용자의 의견이나 반응

UI (사용자 인터페이스) 사용자가 보고 누르는 화면 디자인

반응형 디자인 컴퓨터, 태블릿, 스마트폰 모두에서 잘 보이는 디자인

자동화 반복 작업을 컴퓨터가 자동으로 처리하게 만드는 것

API 프로그램끼리 정보를 주고받는 통로 (택배 배송 창구 같은 것)

결제 연동 웹사이트에 카드 결제 기능을 추가하는 것

용어	뜻
디지털 노마드	노트북 하나로 여행하면서 일하는 사람
커뮤니티	같은 관심사를 가진 사람들의 모임
변아웃	과로로 인해 의욕과 에너지가 완전히 소진된 상태

D. 이 책에서 소개한 인물

Pieter Levels (피터 레벨스)

- **국적:** 네덜란드
- **시작:** 2014년, 부모님 집에서
- **대표 서비스:** NomadList, RemoteOK, PhotoAI
- **현재 수입:** 연 약 38억 원
- **특징:** 직원 없이 혼자 운영
- **SNS:** x.com/levelsio

Marc Lou (마크 루)

- **국적:** 프랑스
- **시작:** 27번 실패 후 성공
- **대표 서비스:** ShipFast, 다수의 마이크로 SaaS
- **특징:** 작은 제품을 빠르게 만드는 전략
- **SNS:** x.com/marc_louvion

이 부록은 필요할 때마다 돌아와서 참고하세요.

여러분의 바이브 코딩 여정에 작은 나침반이 되기를 바랍니다.