

AITS ^{3rd} 교사자율연수

온라인 세미나 LIVE

주제: 바이브 코딩을 위한 Git & GitHub
완벽 가이드



일시: 2026년 5월 9일(토) 21:00~22:30

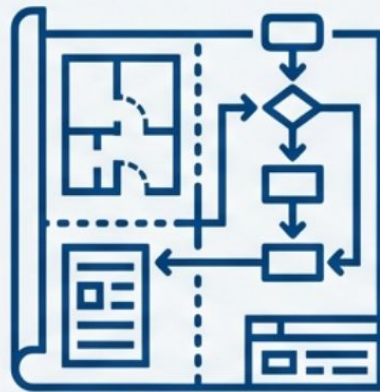
강사: 대전둔천초등학교 김진관 선생님

바이브 코딩의 정의

개발자가 생성형 AI의 도움을 받아 직감(Vibe)과 논리를 결합하여 코드를 작성하는 새로운 행위.



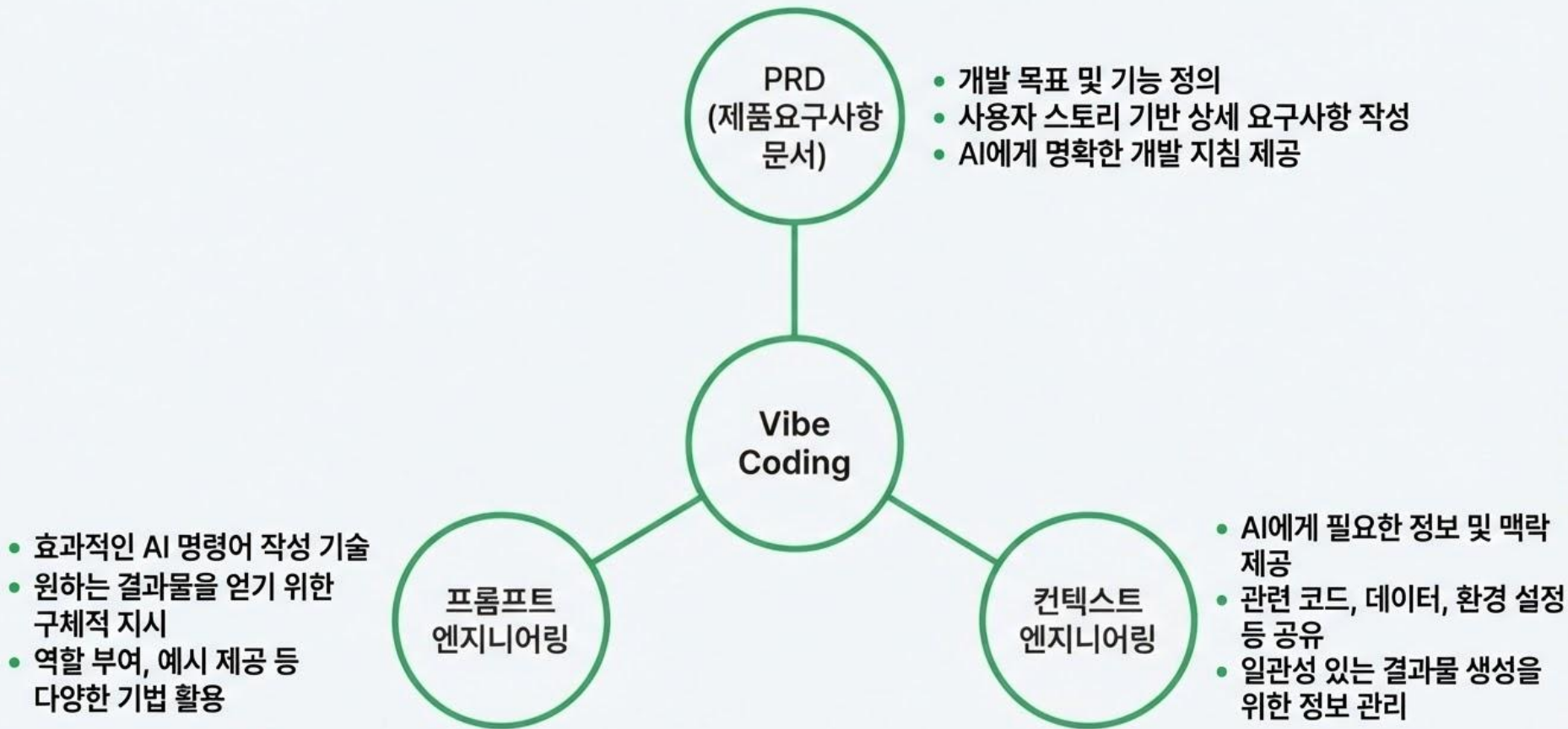
Old Syntax



New Structure

핵심: 단순한 '느낌'이 아닌, AI를 통제하는 '설계 능력'

바이브 코딩의 3대 핵심 요소



기술적 기초: 웹앱의 구성 요소

HTML



역할: 구조와 뼈대
비유: 집의 뼈대와 벽

CSS



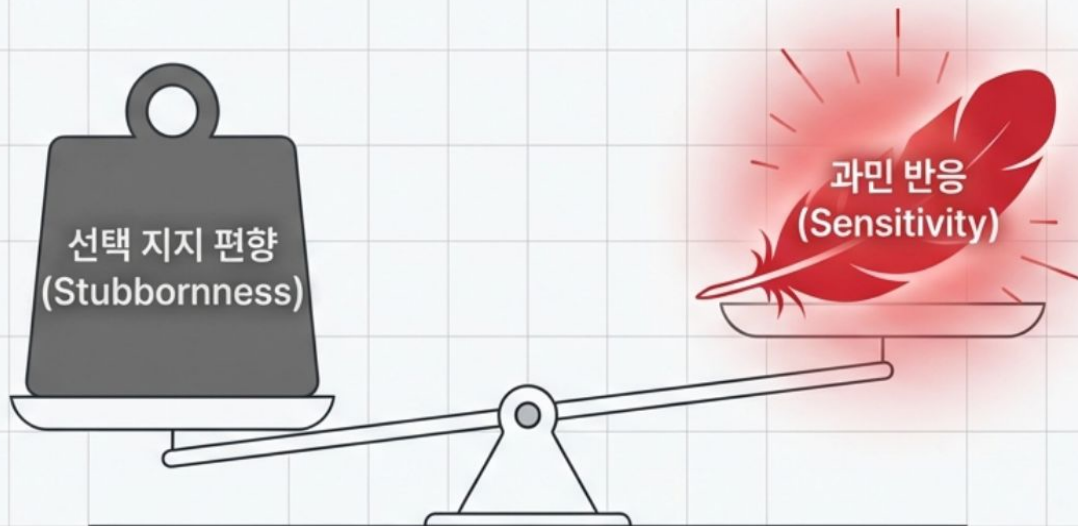
역할: 스타일과 디자인
비유: 인테리어, 가구

JavaScript



역할: 동적 기능
비유: 문 열기, 전등 켜기

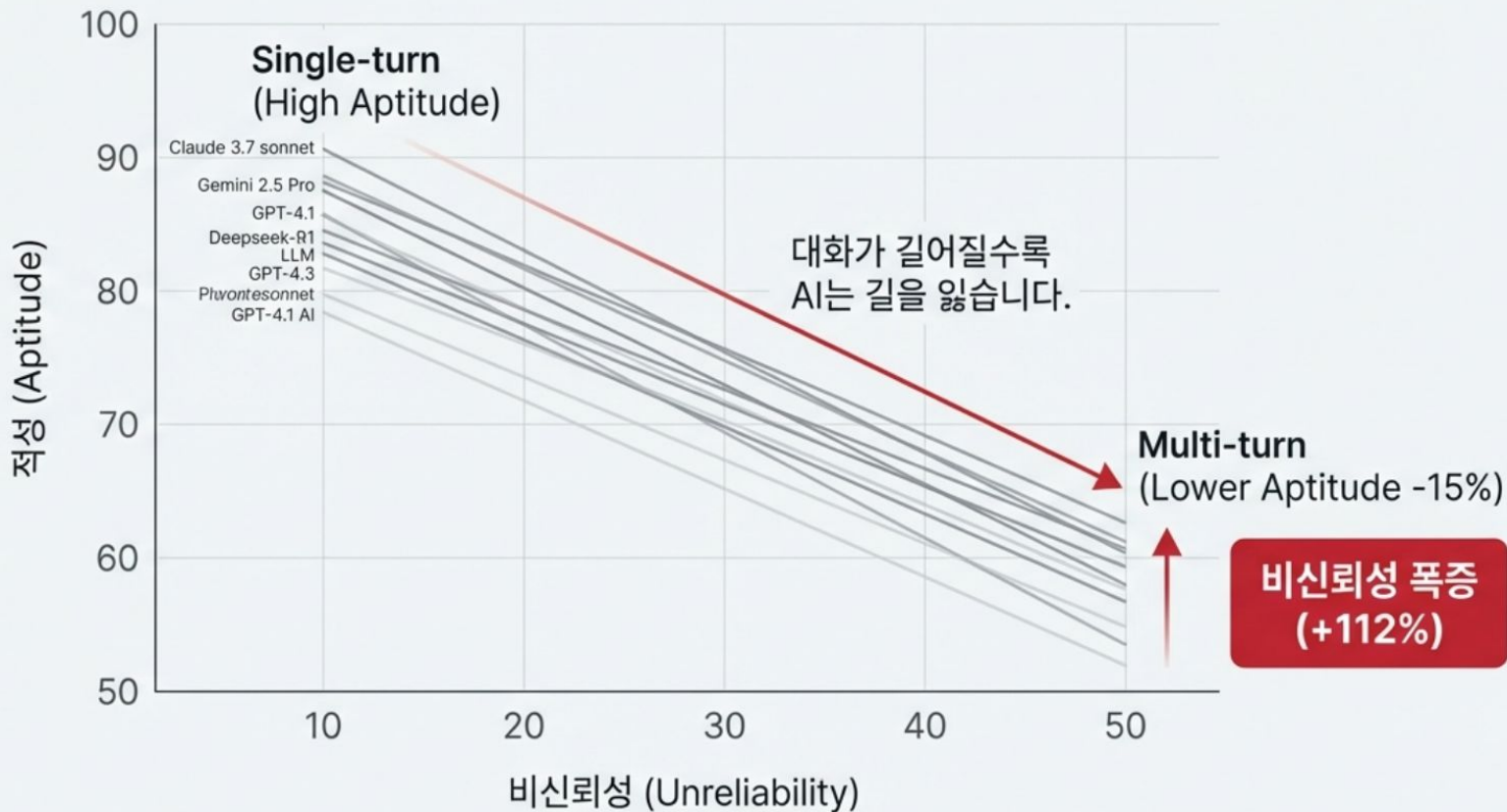
LLM의 심리학: 고집과 과민 반응



자신의 초기 답변을
고수하여 일관성을
유지하려 함.

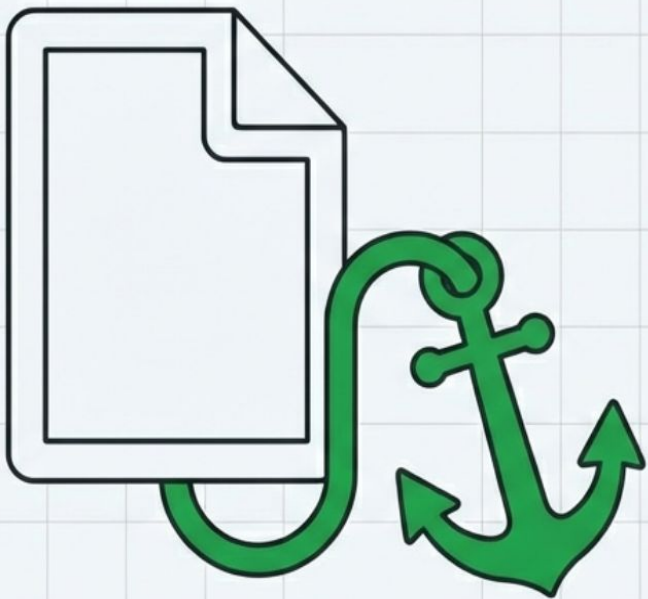
반대 조언에 대해
이상적인 모델보다 **2.58배**
더 과도하게 반응함.

다중 턴 대화의 위험성



솔루션 1: PRD (제품요구사항 문서)

AI를 위한 장기 설명서이자 기준점



1. 문제 정의
(Problem Definition)
2. 핵심 기능 목록
(Core Features)
3. 제약사항 및 성공 기준
(Constraints)

AI가 길을 잃지 않도록 붙잡아주는 '닻'의 역할

솔루션 2: 프롬프트 엔지니어링

자연어의 구조화 및 구체화



Bad: 퀴즈 앱 만들어줘.



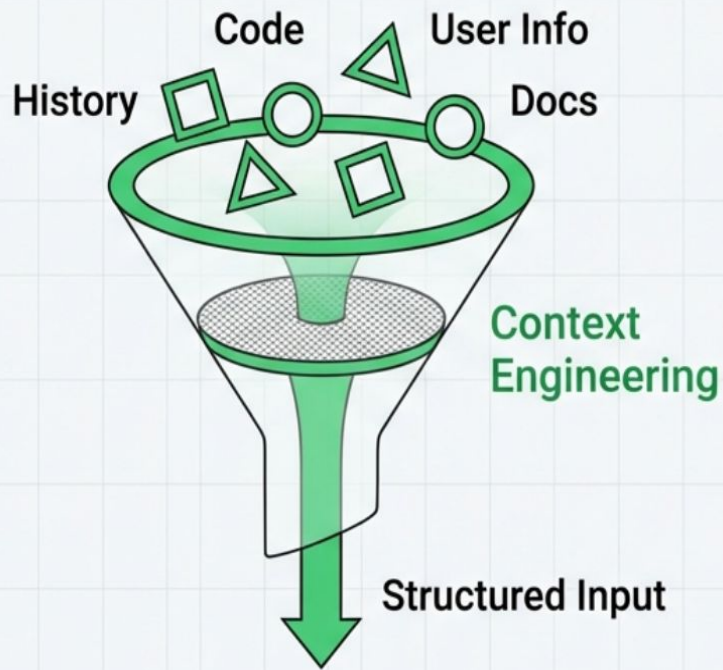
구체화 (Specification)



Good: 4지선다 객관식 퀴즈 앱, 문제은행 20문항, 점수와 정답 피드백 화면을 포함해서 만들어줘.

솔루션 3: 컨텍스트 엔지니어링

AI를 위한 정보 설계

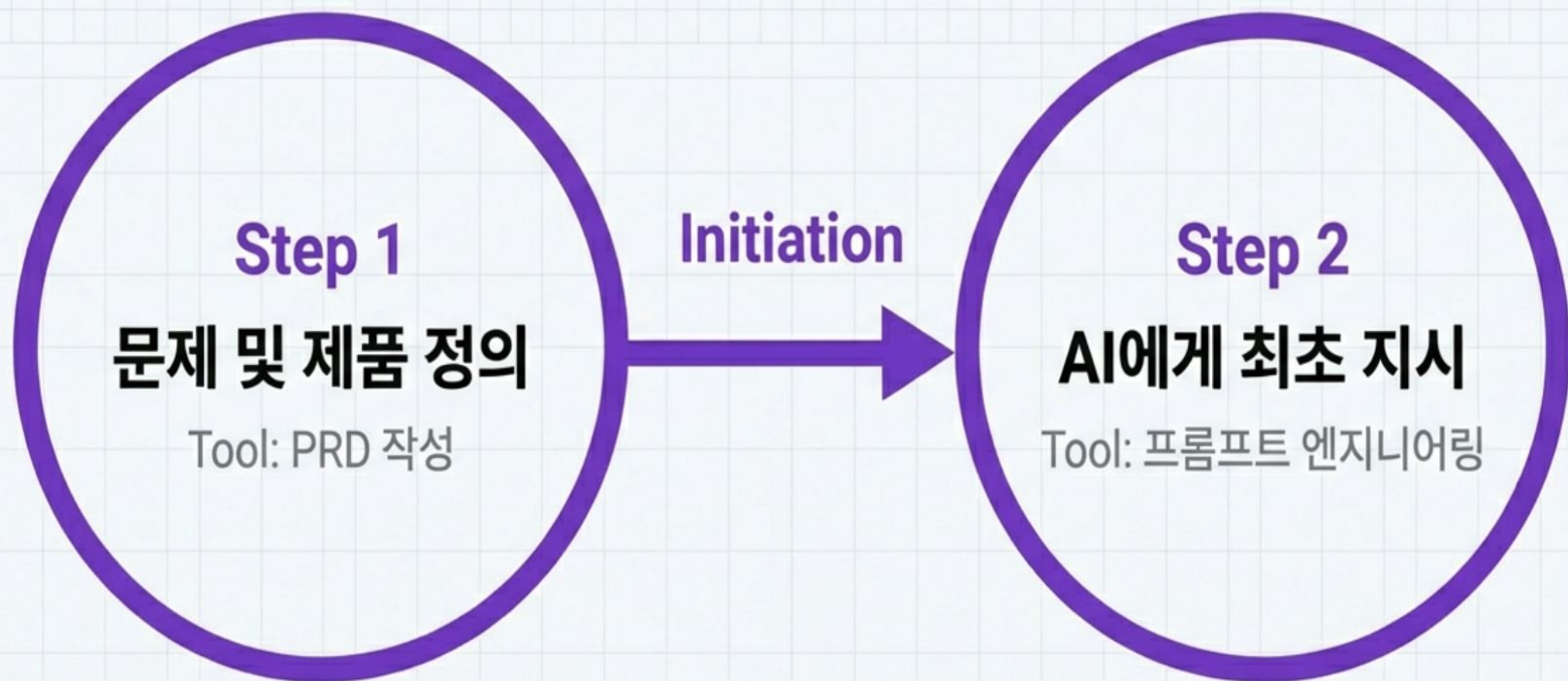


- 대화 히스토리 정리
- 이전 코드 및 수정 내역 통합
- 맥락의 재설정 (Resetting)

바이브 코딩 방법론 요약

요소 (Element)	역할 (Role)	비유 (Metaphor)
PRD	목표 (Goal)	기준점 (Anchor)
Prompt	지시 (Instruction)	핸들링 (Steering)
Context	환경 (Environment)	정보 설계 (Architecture)

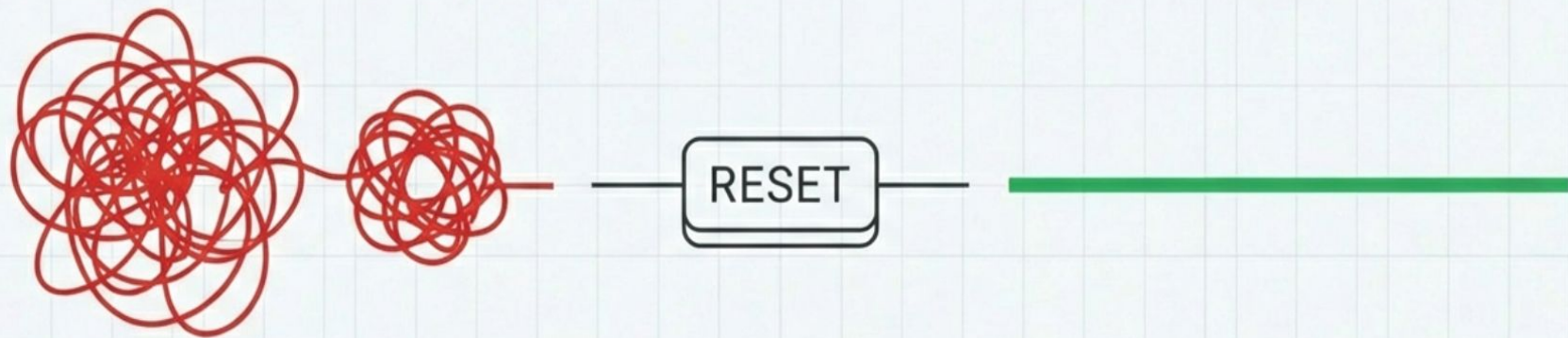
바이브 코딩 프로세스: 시작 단계



바이브 코딩 프로세스: 반복과 개선



전략: 길을 잃었다면 '새로운 출발'을 하라



Multi-Turn Confusion

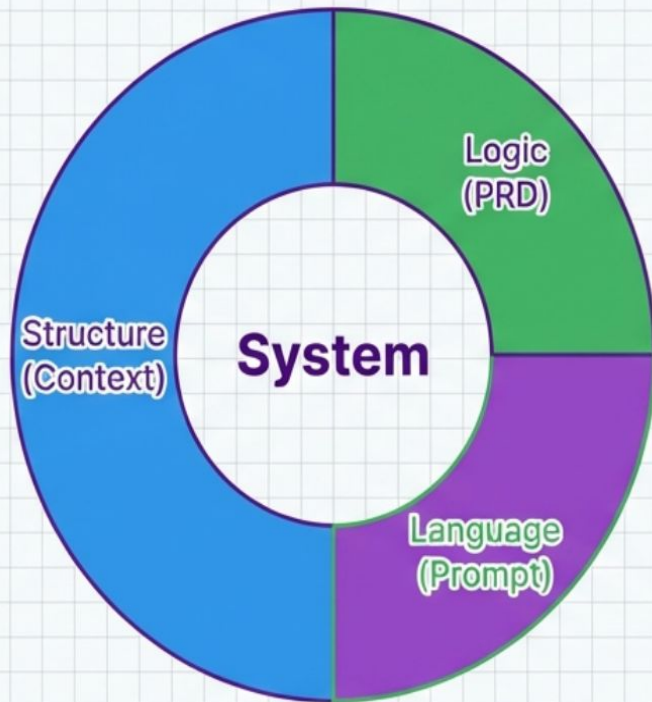
Single-Turn Clarity

“일관성을 잃은 AI와 논쟁하지 마십시오. 대화창을 초기화하십시오.”

바이브 코딩 학습 로드맵



결론: 마법이 아닌, 체계적인 시스템



바이브 코딩은 인간의 직관을 AI의 연산 능력과 결합하여,
가장 효율적으로 소프트웨어를 구축하는 현대의 공학입니다.

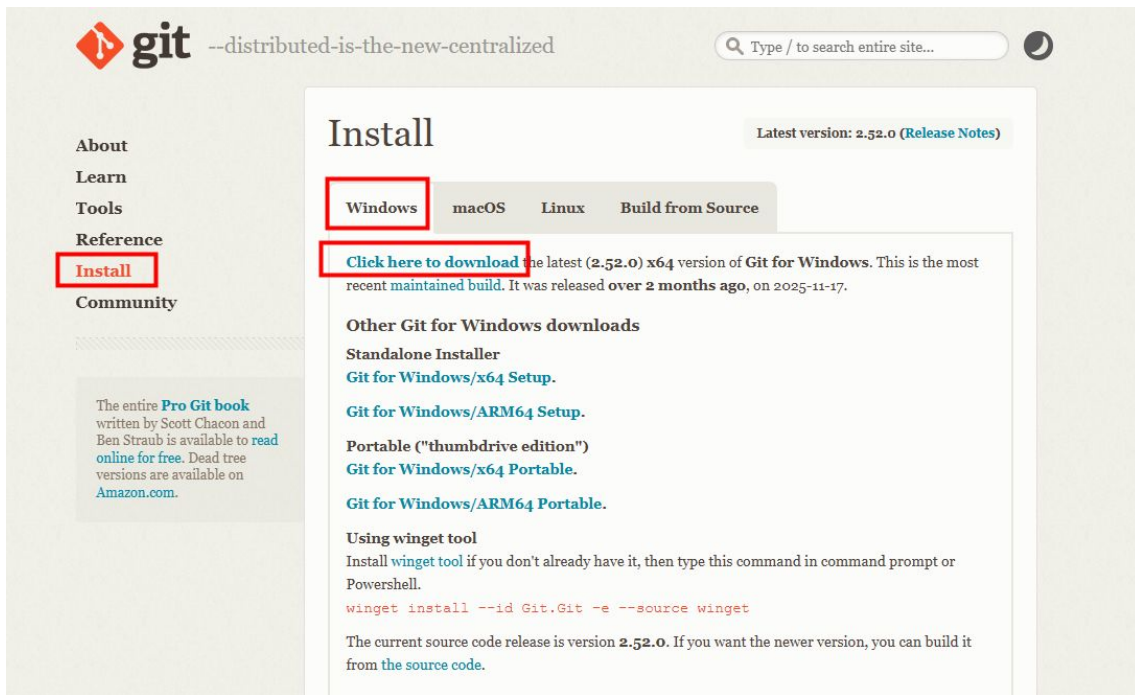
깃허브(GitHub) 시작하기: 회원가입부터 데스크탑 활용까지

비전공자와 초보자를 위한 가장 친절한 가이드



START WITH THE ESSENTIALS

[Windows] 깃(Git) 다운로드 및 설치방법



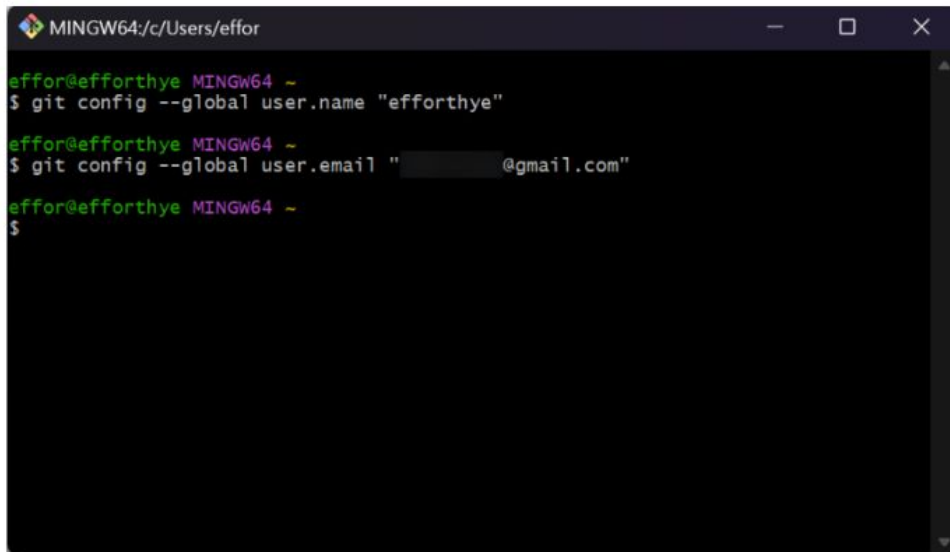
The screenshot shows the Git website's 'Install' page. The page title is 'Install' and it indicates the latest version is 2.52.0. There are four tabs: 'Windows', 'macOS', 'Linux', and 'Build from Source'. The 'Windows' tab is selected and highlighted with a red box. Below the tabs, there is a link 'Click here to download' which is also highlighted with a red box. The text below the link states: 'the latest (2.52.0) x64 version of Git for Windows. This is the most recent maintained build. It was released over 2 months ago, on 2025-11-17.' Below this, there are sections for 'Other Git for Windows downloads' including 'Standalone Installer', 'Git for Windows/x64 Setup.', 'Git for Windows/ARM64 Setup.', 'Portable ("thumbdrive edition")', and 'Git for Windows/x64 Portable.' There is also a section for 'Using winget tool' with instructions on how to install it and a command: `winget install --id Git.Git -e --source winget`. The page also mentions the current source code release is version 2.52.0 and provides a link to the source code.

<https://git-scm.com/install/windows>

<https://m.blog.naver.com/efforthye/223042611627>

[Windows] 깃(Git) 다운로드 및 설치방법

```
git config --global user.name "자신의Git사용자이름"  
git config --global user.email "자신의Git이메일@xxxxxx.xxx"
```



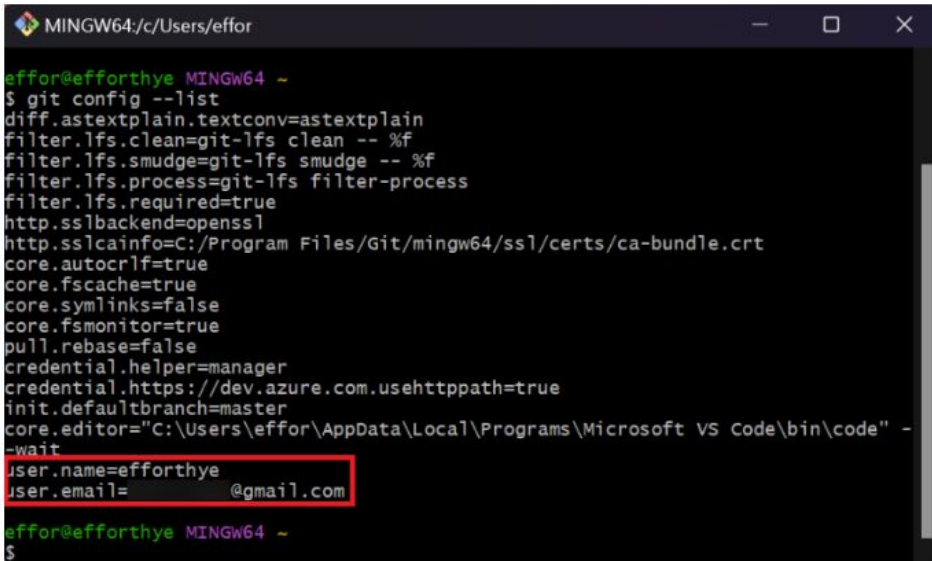
A screenshot of a terminal window titled 'MINGW64:/c/Users/effor'. The prompt is 'effor@efforthye MINGW64 ~'. The user enters the command '\$ git config --global user.name "efforthye"'. The prompt changes to 'effor@efforthye MINGW64 ~' and the user enters '\$ git config --global user.email " [redacted] @gmail.com"'. The prompt changes to 'effor@efforthye MINGW64 ~' and the user enters '\$'. The prompt changes to 'effor@efforthye MINGW64 ~' and the user enters '\$'.

Git Bash에 해당 명령어를 입력하여
사용중인 자신의 Git 사용자 이름과
Git 이메일을 등록하여 줍니다.

<https://m.blog.naver.com/efforthye/223042611627>

[Windows] 깃(Git) 다운로드 및 설치방법

```
git config --list
```

A terminal window titled 'MINGW64:c/Users/effor' showing the output of the 'git config --list' command. The output lists various Git configuration settings. The last two lines, 'user.name=efforthye' and 'user.email=@gmail.com', are highlighted with a red rectangular box.

```
effor@efforthye MINGW64 ~  
$ git config --list  
diff.astextplain.textconv=astextplain  
filter.lfs.clean=git-lfs clean -- %f  
filter.lfs.smudge=git-lfs smudge -- %f  
filter.lfs.process=git-lfs filter-process  
filter.lfs.required=true  
http.sslbackend=openssl  
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt  
core.autocrlf=true  
core.fscache=true  
core.symlinks=false  
core.fsmonitor=true  
pull.rebase=false  
credential.helper=manager  
credential.https://dev.azure.com.usehttppath=true  
init.defaultbranch=master  
core.editor="C:\Users\effor\AppData\Local\Programs\Microsoft VS Code\bin\code" -  
-wait  
user.name=efforthye  
user.email=@gmail.com  
effor@efforthye MINGW64 ~  
$
```

Git 사용자 등록이 정상적으로
완료되었는지 아래의 명령어를 입력하여
확인해봅니다.

<https://m.blog.naver.com/efforthye/223042611627>

Build and ship software on a single, collaborative platform

Join the world's most widely adopted AI-powered developer platform.

Enter your email

Sign up for GitHub

Try GitHub Copilot

<https://github.com/> 접속 - 회원가입

Already have an account? [Sign in](#) →

Create your free account

Explore GitHub's core features for individuals and organizations.

See what's included ▾



Sign up for GitHub

Email*



Password*

Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Username*

Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.

Your Country/Region*



For compliance reasons, we're required to collect country information to send you occasional updates and announcements.

Create account >

By creating an account, you agree to the [Terms of Service](#). For more

이메일 주소, 비밀번호, 사용자 이름(중복X), 국가 입력 => 계정 생성하기

Already have an account? [Sign in](#) →

Create your free account

Explore GitHub's core features for individuals and organizations.

See what's included ▾



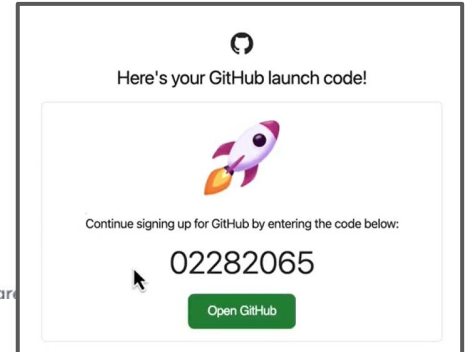
Confirm your email address

We have sent a code to leetaewons@naver.com

Enter code

Continue

Didn't get your email? [Resend the code](#) or [update your email address](#).



By creating an account, you agree

입력한 이메일 확인하여 코드 입력하기



Sign in to GitHub

Your account was created successfully! ×
Please sign in to continue.


Username or email address

Password

[Forgot password?](#)

Sign in

or

 Continue with Google

New to GitHub? [Create an account](#)

[Sign in with a passkey](#)

가입한 정보 입력해주고 로그인하기

Public profile

Account

Appearance

Accessibility

Notifications

Access

Billing and licensing

Emails

Password and authentication

Sessions

SSH and GPG keys

Organizations

Enterprises

Public profile

Name

Your name may appear around GitHub where you contribute or are mentioned. You can remove it at a time.

Public email

Select a verified email to display

You can manage verified email addresses in your [email settings](#)

Bio

You can @mention other users and organizations to link to them.

프로필 입력하기

The image shows the GitHub user settings page, specifically the 'Appearance' section. The 'Appearance' menu item is highlighted with a red box. Below it, the 'Theme mode' dropdown is also highlighted with a red box and contains the text 'Sync with system'. To the right of this dropdown, a note states: 'GitHub theme will match your system active settings'. Below the settings, there are two preview cards: 'Light theme' and 'Dark theme'. The 'Dark theme' card has an 'Active' button in its top right corner. Each preview card shows a mockup of the GitHub interface in that theme, with a green progress bar and a red and green status indicator.

Account

Appearance

Accessibility

Notifications

Access

Billing and licensing

Emails

Password and authentication

Sessions

SSH and GPG keys

Organizations

Enterprises

Moderation

Code, planning, and automation

Repositories

Codespaces

Theme mode

Sync with system

GitHub theme will match your system active settings

Light theme

This theme will be active when your system is set to "light mode"

Light default

Dark theme

This theme will be active when your system is set to "dark mode"

Active

Dark default

Appearance - 테마모드 설정

Git (깃)



내 컴퓨터의 타임머신 (Local)

버전 관리 및 변경 이력 추적

GitHub (깃허브)



클라우드 저장소 (Cloud)

협업, 공유, 그리고 백업

Git은 기록하고, GitHub는 공유합니다.

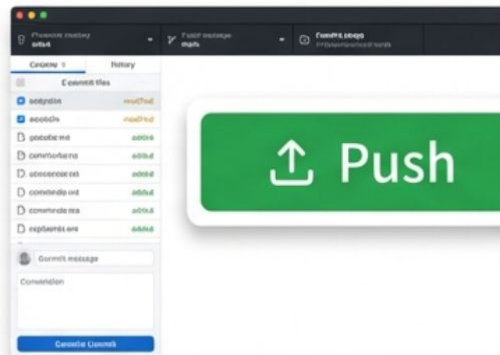
Why GitHub Desktop?

CLI (명령어 방식)

```
$ git commit -m "update"  
$ git push origin main
```

- 복잡한 텍스트 명령
- 높은 진입 장벽

GUI (데스크탑 앱)



- 직관적인 버튼
- 드래그 앤 드롭
- 시각적 확인

명령어 암기 없이, 클릭으로 시작하는 버전 관리.

Step 1: 계정 생성 및 보안



가입 (Sign Up)
github.com 접속



인증 (Verify)
이메일 코드 입력



2단계 인증 (2FA)
필수 보안 설정

중요: 모바일 앱(OTP) 또는
SMS 인증이 필수입니다.

Step 2: 설치 및 로그인



설치 후 로그인을 해야
저장소 접근 가능

- 1. **다운로드:**
desktop.github.com ↓
- 2. **설치:**
Windows / Mac 자동 감지 📱
- 3. **계정 연동:**
File > Options > Sign in to
GitHub.com 👤

Download GitHub Desktop

Focus on what matters instead of fighting with Git. Whether you're new to Git or a seasoned user, GitHub Desktop simplifies your development workflow.

[Download for Windows \(64bit\)](#)

Try beta features and help improve future releases

Experience the latest features and bug fixes before they're released.

[Check out Beta](#)

Prefer the MSI?

Need to download the package to install across your organization?

[Download for Windows \(MSI\)](#)

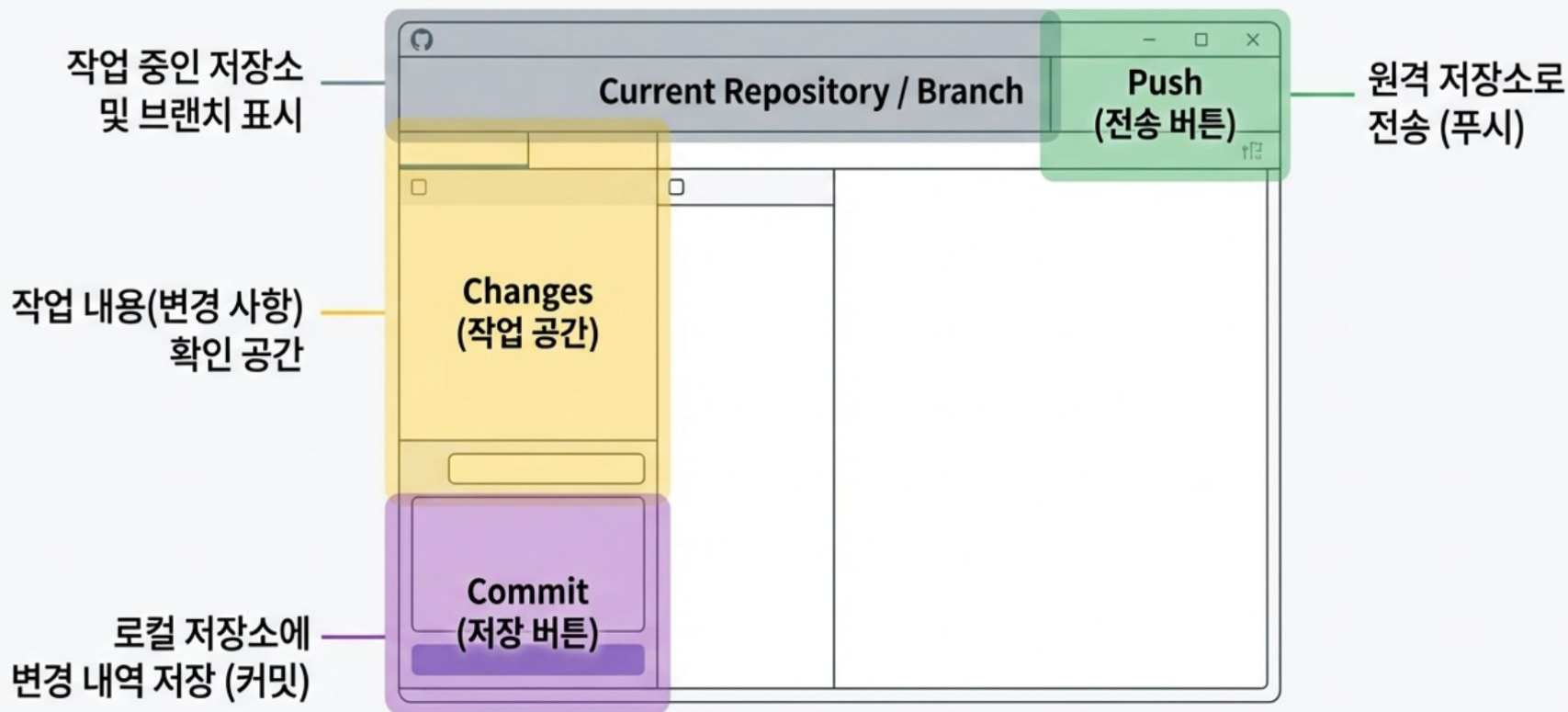
Mac?

Need to download for macOS?

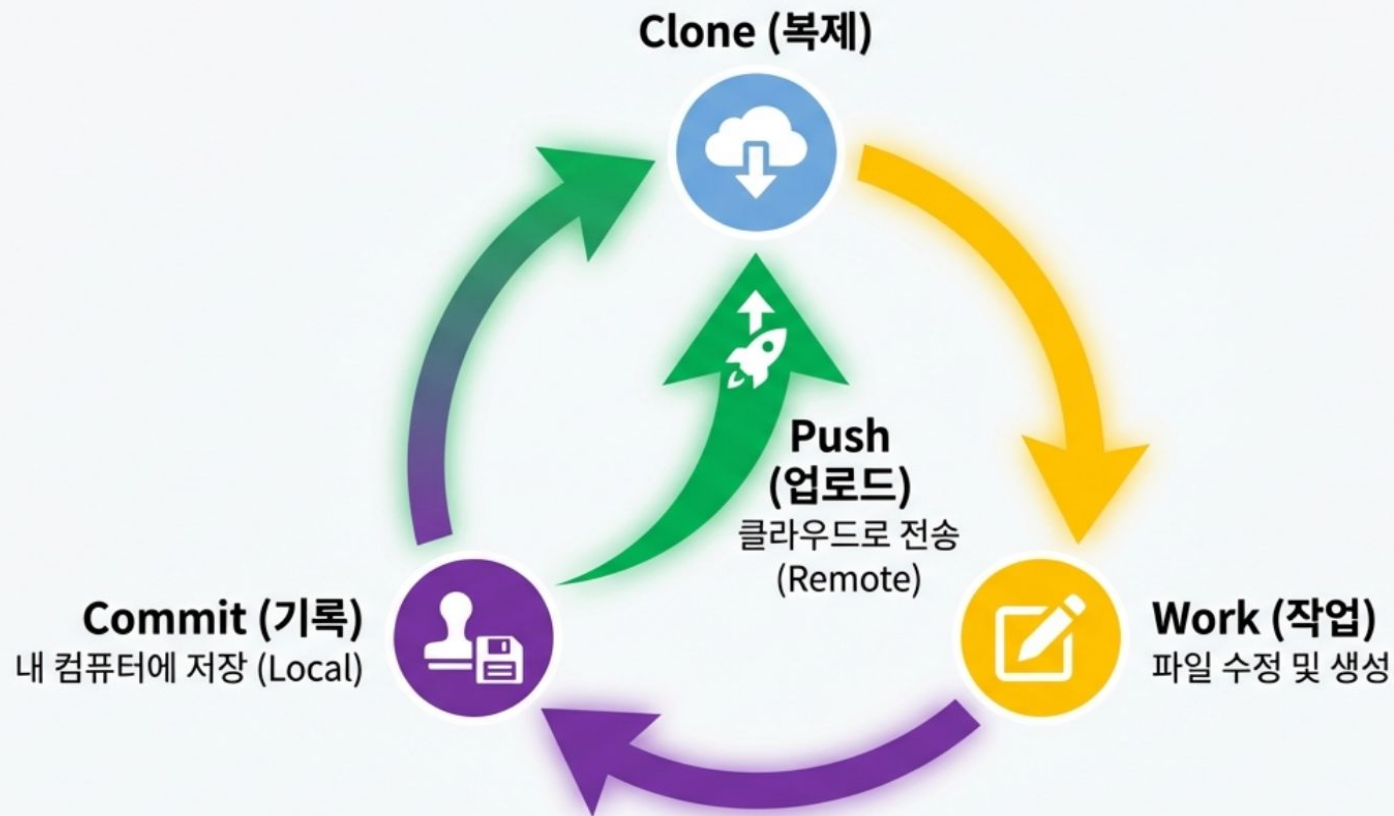
[Download for macOS](#)

By downloading, you agree to the [Open Source Applications Terms](#).

인터페이스 구조 (Interface Anatomy)



The Golden Cycle: 작업의 순환



Action 1: Clone (프로젝트 가져오기)

New (새 프로젝트)



File > New Repository

Clone (기존 프로젝트)



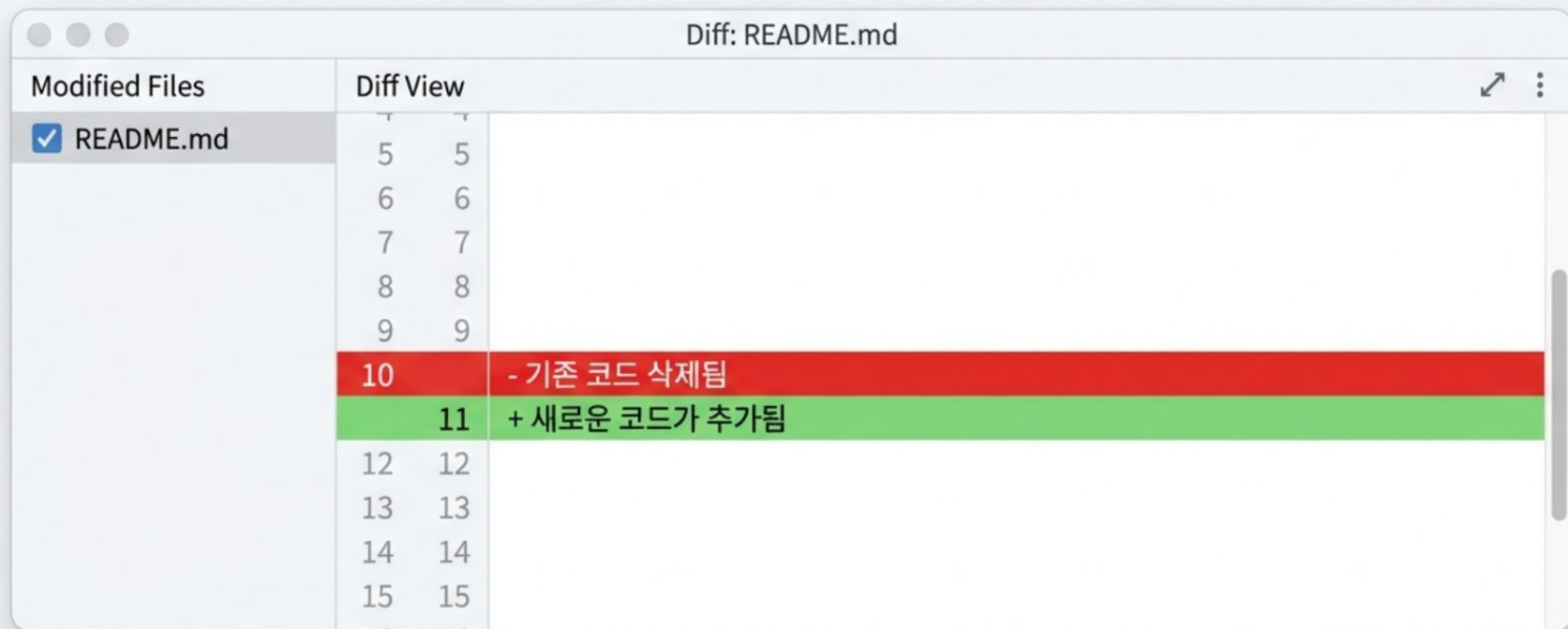
File > Clone Repository

Repository URL:

Local Path:



Action 2: Changes (변경 사항 확인)



수정한 내용을 눈으로 직접 확인하세요.

Action 3: Commit (로컬 저장)

Summary (필수)

Description (선택)

Commit to main



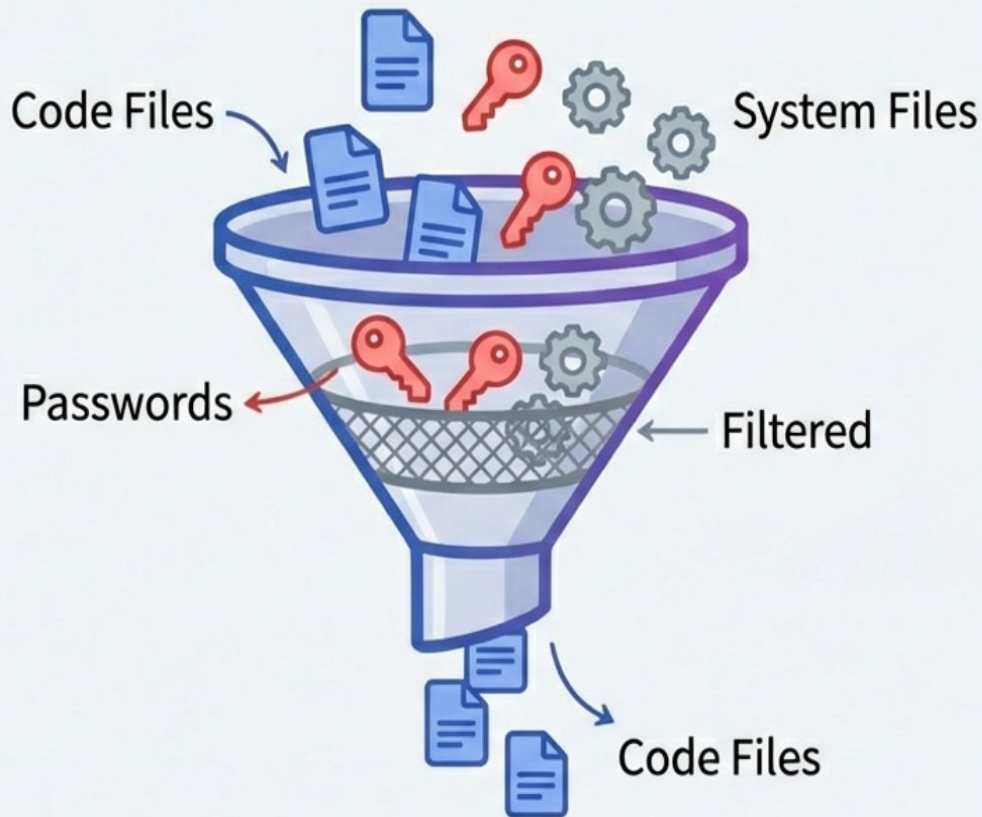
주의: 아직 인터넷에 올라가지 않았습니다.
내 컴퓨터에만 저장된 상태!

Action 4: Push (클라우드 업로드)



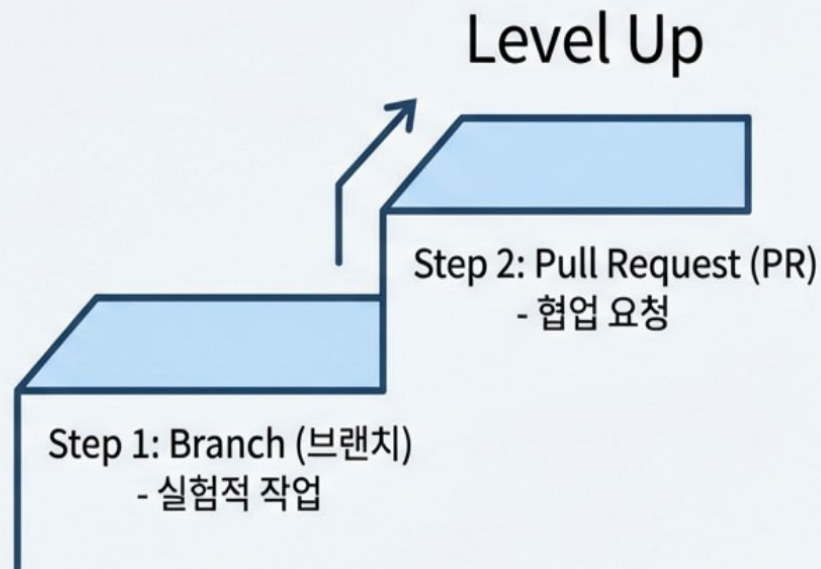
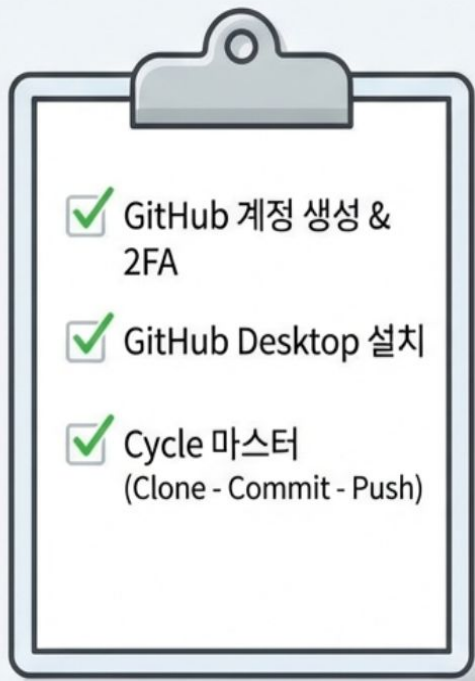
Commit된 내용을 GitHub 서버로 전송합니다.
Push 후 웹사이트에서 History를 꼭 확인하세요.

Essential: .gitignore (파일 필터링)



- 보안 파일, 대용량 파일 등 업로드를 막아야 하는 목록
- 설정 방법: Repository Settings > Ignored files

요약 및 다음 단계 (Next Steps)



이제 여러분의 코드를 전 세계와 공유하세요.

