

객체지향 프로그래밍 응용 및 버전 관리

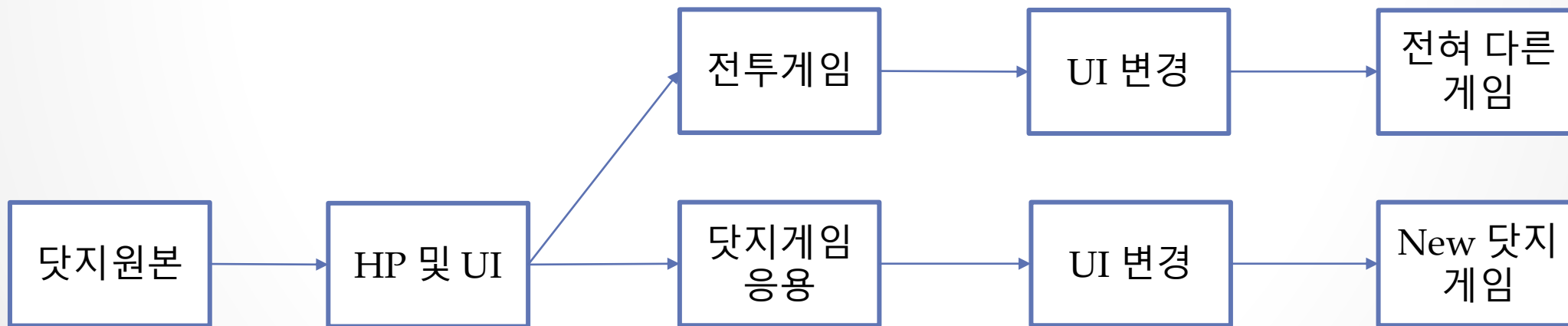
이준

목표

- 다양한 응용을 하면서 관련된 객체 지향 프로그래밍 방법을 배워 보도록 하자!
- 버전 관리 시스템에 대한 기본적인 이해를 해야함
- 플레이어의 체력 관리 및 UI 처리 시도해 보기!

버전 관리

- 지금까지 만들어진 닥지 플레이어 버전은 여기서..
- <https://drive.google.com/file/d/1oRjR5hTYfL4OfwfZys89xMaRNY2ptiD7/view?usp=sharing>
- 여기서 받아서 확인 할 수 있음
- 이후 다음의 응용들을 고려해 볼 수 있음
 - 1. 플레이어의 HP 관리 및 UI 처리
 - 2. 전투 시스템으로 변경 (적 HP 처리)
 - 3. 3D 모델링된 애니메이션 정보 사용 및 응용

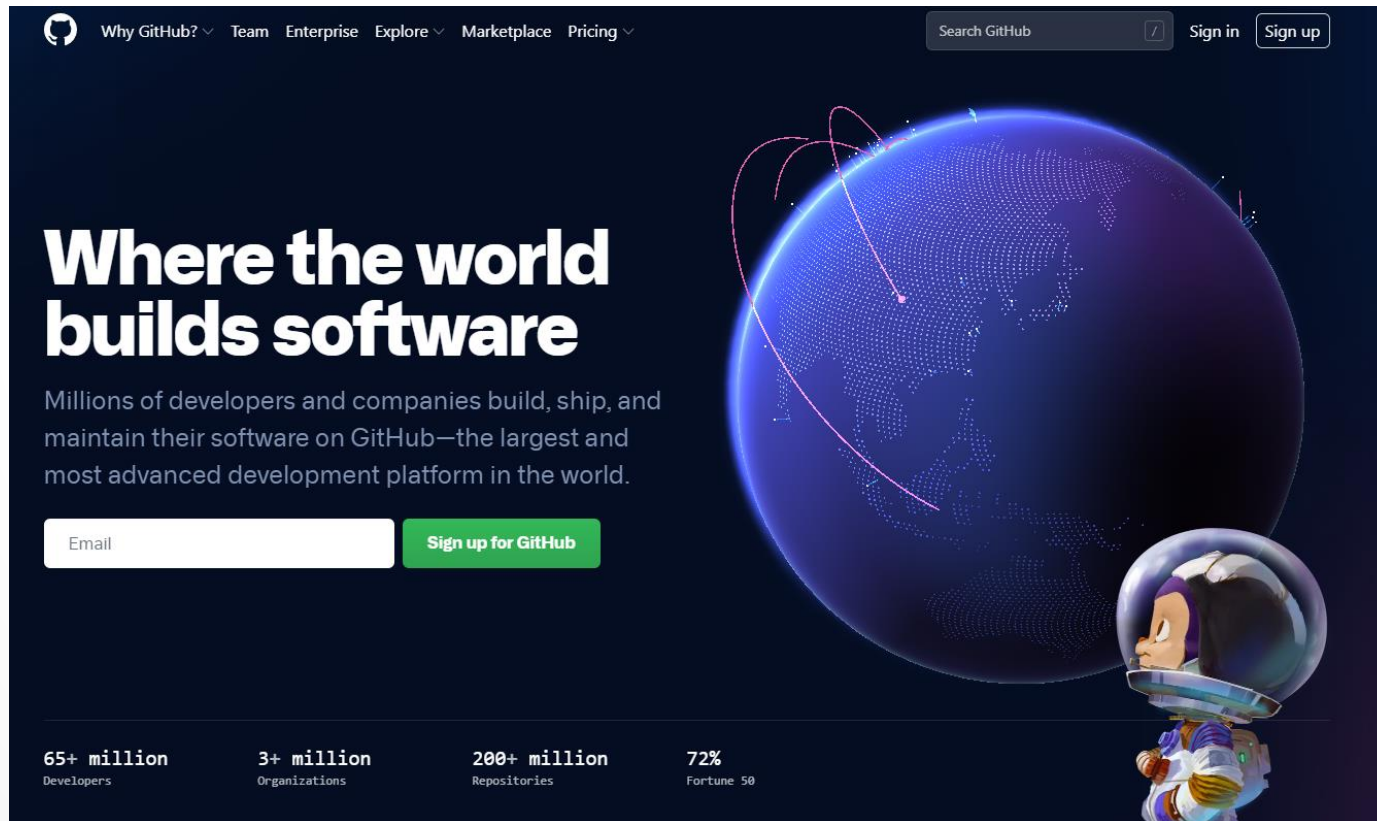


버전 관리

- 공동 작업을 수행하는 경우..
- 모든 팀원들이 구글 드라이브만으로 공유하기가 쉽지 않음...
- EX : UI 작업, 몬스터작업이 동시에 이루어지는 경우.. (구글 드라이브 업데이트를 잘못 하면 버전이 갈려 버리는 상황이 생김)
- 버전 관리할 수 있는 시스템이 반드시 필요함

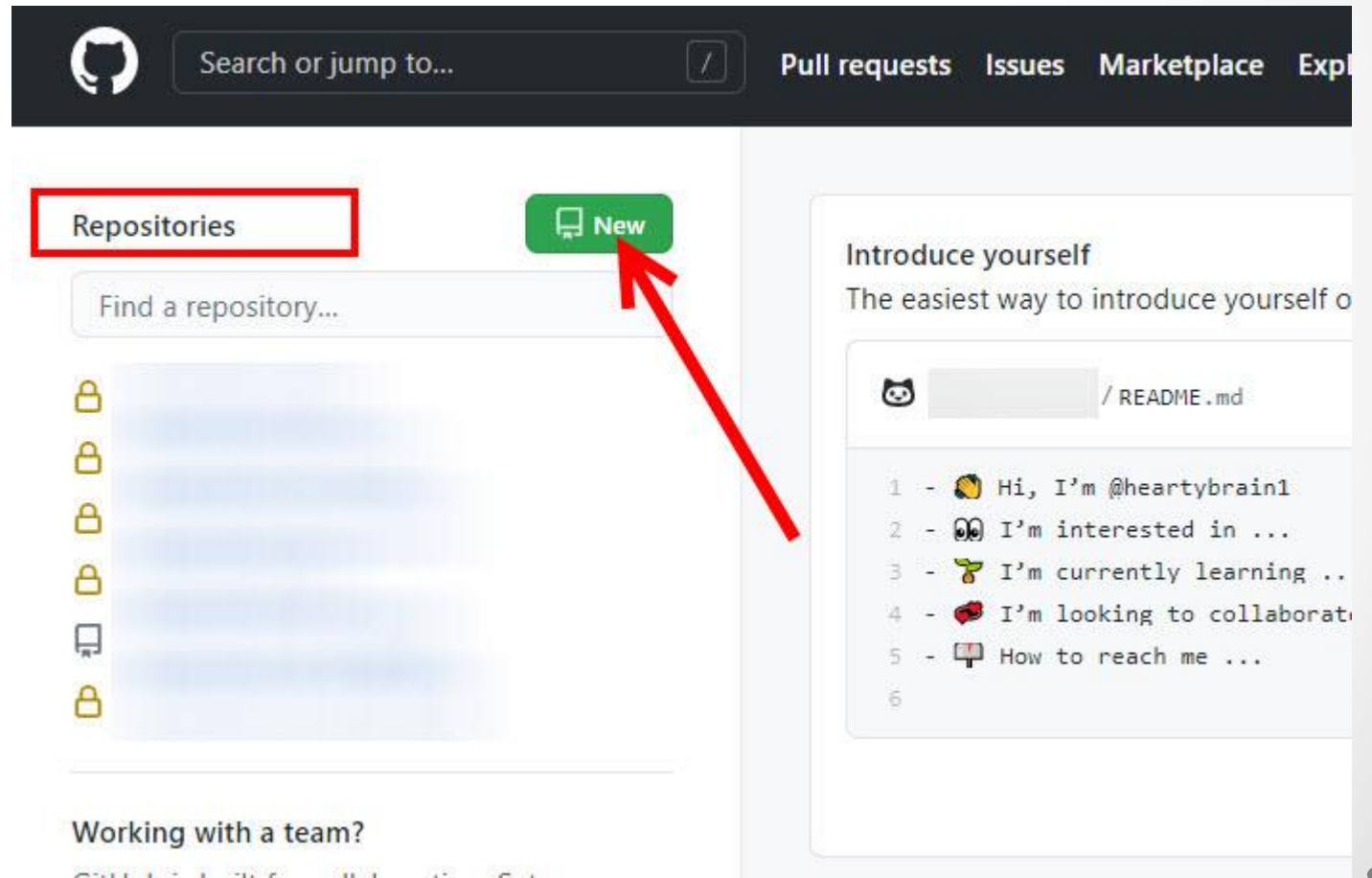
버전 관리

- 깃 허브 (Git Hub) <https://github.com/>
- 버전관리를 할 수 있는 웹 호스팅 서비스 (세계 최대 규모!)
- Git 이라는 프로그램을 사용하여 관리할 수 있음 (공개 / 비공개 모드 지원)
- 회원 가입을 해야함!



버전 관리

- 깃허브에서 새로운 프로젝트 등록하는 방법 (Repository)
- 로그인 한 다음 New 버튼 클릭



버전

- 리포지토리의 이름(repository name)과 설명(description)을 기입한다.
- private를 선택하면 리포지토리를 공개하지 않는다. (public 으로 선택!)
- 저는 hcijun-dodge 으로 생성하도록 하겠습니다 (여러분들도 고유의 아이디로 만들어 보세요)
- README 파일 추가 클릭 (프로젝트 설명에 필요)
- .gitignore 추가 클릭 (유니티프로젝트로 설정)
- 그후 Create 버튼 클릭!

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository.](#)

Owner *

hcijun ▼

Repository name *

hcijun-dodge ✓

Great repository names are short and memorable. Need inspiration? How about [turbo-parakeet?](#)

Description (optional)

닷지 플레이어 게임입니다



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

☒ Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: Unity ▼

☐ Choose a license


A license tells others what they can and can't do with your code. [Learn more.](#)


This will set `main` as the default branch. Change the default name in your [settings](#).

Create repository

버전 관리


- 생성된 github 의 닷지 플레이어 repository!
- 공개 모드이기 때문에 어디서나 접근이 가능 하다! (원하지 않으면 비공개 모드로 변경할 수 있음)


 Search or jump to... / Pull requests Issues Marketplace Explore



hcijun / hcijun-dodge 


<> Code ! Issues 🔗 Pull requests ▶ Actions 📁 Projects 📖 Wiki 🛡 Security 📈 Insights ⚙ Settings

🔗 main ▾ 🌿 1 branch 🏷 0 tags

Go to file Add file ▾  Code ▾


 hcijun Initial commit 9185e8d now 🕒 1 commit

 .gitignore	Initial commit	now
 README.md	Initial commit	now


README.md 

hcijun-dodge

닷지 플레이어 게임 입니다

About 

닷지 플레이어 게임 입니다

 Readme

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

버전 관리

- README.md 파일 확인해보기
- 프로그램에 대한 설명, 현재 버전의 특징 및 설치 과정등을 설명 해야함
- 추후 라이선스에 대한 부분도 설명을 해줘야 함



Search or jump to...



[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)

[hcijun](#) / [hcijun-dodge](#)

[Unwatch](#)

[Code](#)

[Issues](#)

[Pull requests](#)

[Actions](#)

[Projects](#)

[Wiki](#)

[Security](#)

[Insights](#)

[Settings](#)

[main](#)

[hcijun-dodge](#) / [README.md](#)

[Go to file](#)



hcijun Initial commit

Latest commit 9185e8d 2 minutes ago [History](#)

1 contributor

2 lines (2 sloc) | 52 Bytes

[Raw](#)

[Blame](#)



hcijun-dodge

닷지 플레이어 게임 입니다

버전 관리


- README.md 파일 추가해보기
- 이런 내용을 작성 하면 반드시 Commit Changes를 해야함! (현재 버전의 업데이트를 수행함)

hcijun-dodge / README.md in main Cancel changes

<> Edit file Preview

Spaces 2 Soft wrap

```
1 # hcijun-dodge
2 닷지 플레이어 게임 입니다.
3 재미있는 게임이 되고 싶습니다
```

 Commit changes

Update README.md

Add an optional extended description...

☒ Commit directly to the main branch.
☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes Cancel

버전 관리

- 업데이트 결과 확인해보기!

hcijun / hcijun-dodge

Unwatch

<> Code ! Issues 🔗 Pull requests ▶ Actions 📁 Projects 📖 Wiki 🛡 Security 📈 Insights ⚙ Settings

main

hcijun-dodge / README.md

Go to file

...

hcijun Update README.md

Latest commit 84f3c9a now History

1 contributor

3 lines (3 sloc) 96 Bytes

Raw

Blame



hcijun-dodge

닷지 플레이어 게임 입니다. 재미있는 게임이 되고 싶습니다

버전 관리

- 버전 확인해 보기 커밋을 했기 때문에 업데이트가 된걸 확인해 볼 수 있음!

The screenshot shows a GitHub repository interface. At the top, there are navigation tabs: Actions, Projects, Wiki, Security, Insights, and Settings. Below these, the repository name 'hcijun' is visible, along with 'main' branch, '1 branch', and '0 tags'. There are buttons for 'Go to file', 'Add file', and 'Code'. The commit history table shows two commits: 'Initial commit' for '.gitignore' (9 minutes ago) and 'Update README.md' for 'README.md' (4 minutes ago). The 'README.md' file content is displayed below, showing the title 'hcijun-dodge' and the text '닷지 플레이어 게임 입니다. 재미있는 게임이 되고 싶습니다'. A red box highlights the commit history table.

File	Commit Message	Commit Hash	Time
.gitignore	Initial commit	84f3c9a	9 minutes ago
README.md	Update README.md	84f3c9a	4 minutes ago

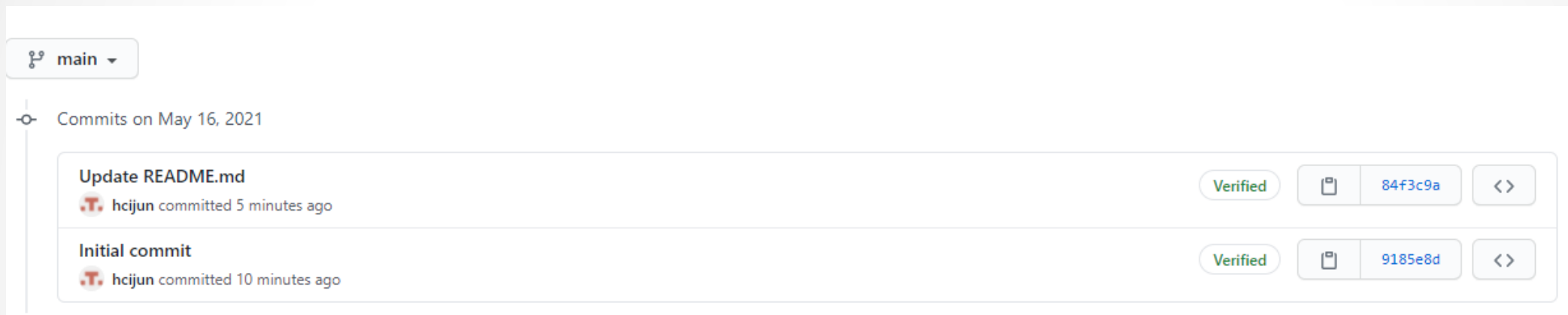
README.md

hcijun-dodge

닷지 플레이어 게임 입니다. 재미있는 게임이 되고 싶습니다

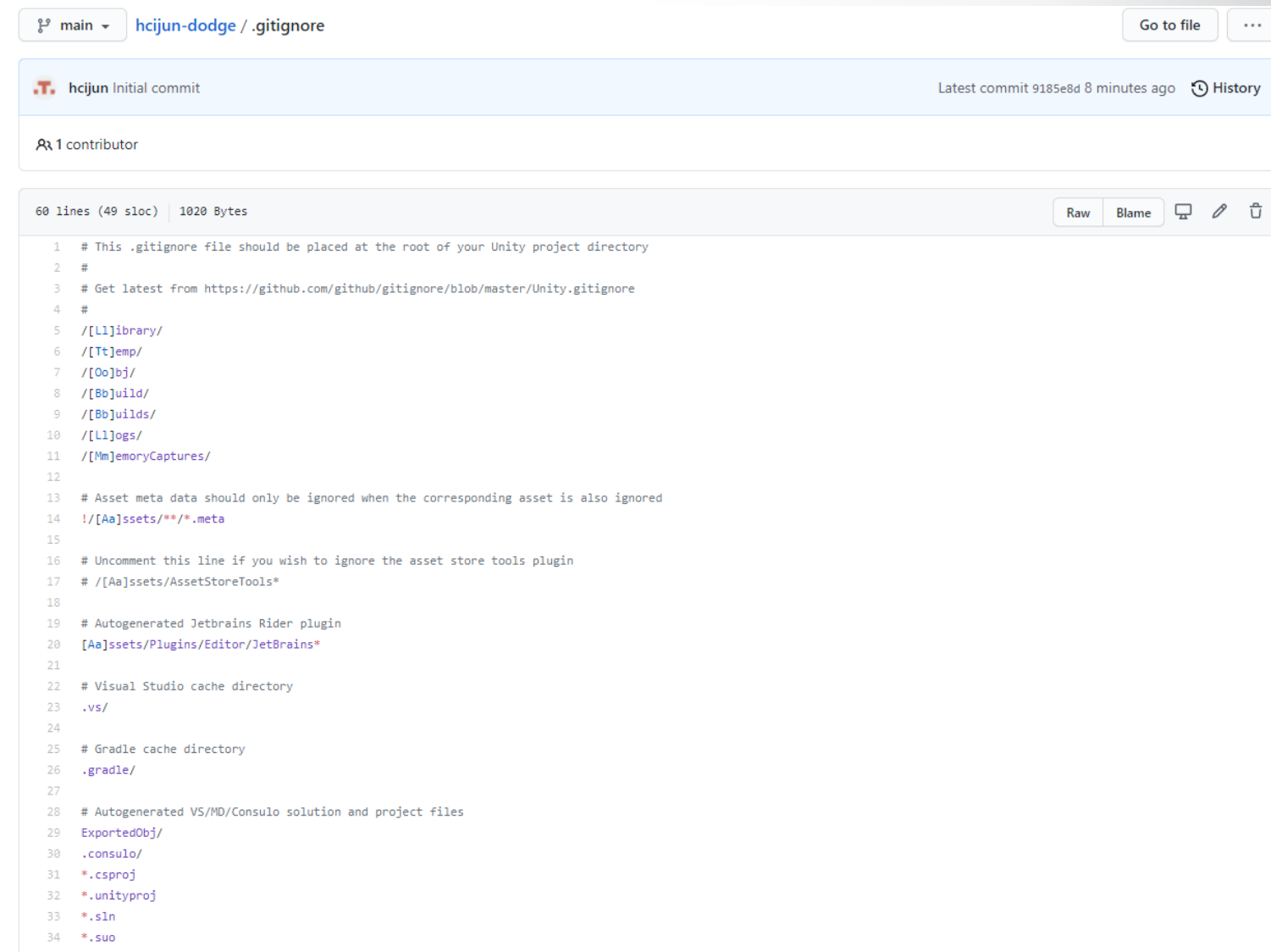
버전 관리

- 누가 언제, 어떤 내용으로 업데이트(commit)을 했는지를 알 수 있음
- 혹시나 버전이 잘못되면 다시 이전으로 되돌릴수도 있음!



버전 관리

- 유니티 프로젝트를 이제 깃허브에 올려 보자
- 고려할점!
 - 인터넷을 통해서 다운받으므로 모든 파일들을 업로드 할 필요가 없음!
 - 특히 유니티에서 생성하는 library, temp, 기타 임시적인 파일 및 폴더를 추적해서 제외하도록 설정 할 수 있음
 - .gitignore 파일에서 설정 되어 있음 – 이 파일은 유니티 프로젝트의 디렉토리의 루트에 위치 해야함



The screenshot shows a GitHub repository for 'hcijun-dodge/.gitignore'. The repository is at the 'main' branch. The latest commit is 'Initial commit' by 'hcijun' 8 minutes ago. The file '.gitignore' is shown with 60 lines (49 sloc) and 1020 Bytes. The content of the file is as follows:

```
1 # This .gitignore file should be placed at the root of your Unity project directory
2 #
3 # Get latest from https://github.com/github/gitignore/blob/master/Unity.gitignore
4 #
5 /[Ll]ibrary/
6 /[Tt]emp/
7 /[Oo]bj/
8 /[Bb]uild/
9 /[Bb]uilds/
10 /[Ll]ogs/
11 /[Mm]emoryCaptures/
12
13 # Asset meta data should only be ignored when the corresponding asset is also ignored
14 !/[Aa]ssets/**/*.meta
15
16 # Uncomment this line if you wish to ignore the asset store tools plugin
17 # /[Aa]ssets/AssetStoreTools*
18
19 # Autogenerated JetBrains Rider plugin
20 [Aa]ssets/Plugins/Editor/JetBrains*
21
22 # Visual Studio cache directory
23 .vs/
24
25 # Gradle cache directory
26 .gradle/
27
28 # Autogenerated VS/MD/Consulo solution and project files
29 ExportedObj/
30 .consulo/
31 *.csproj
32 *.unityproj
33 *.sln
34 *.suo
```

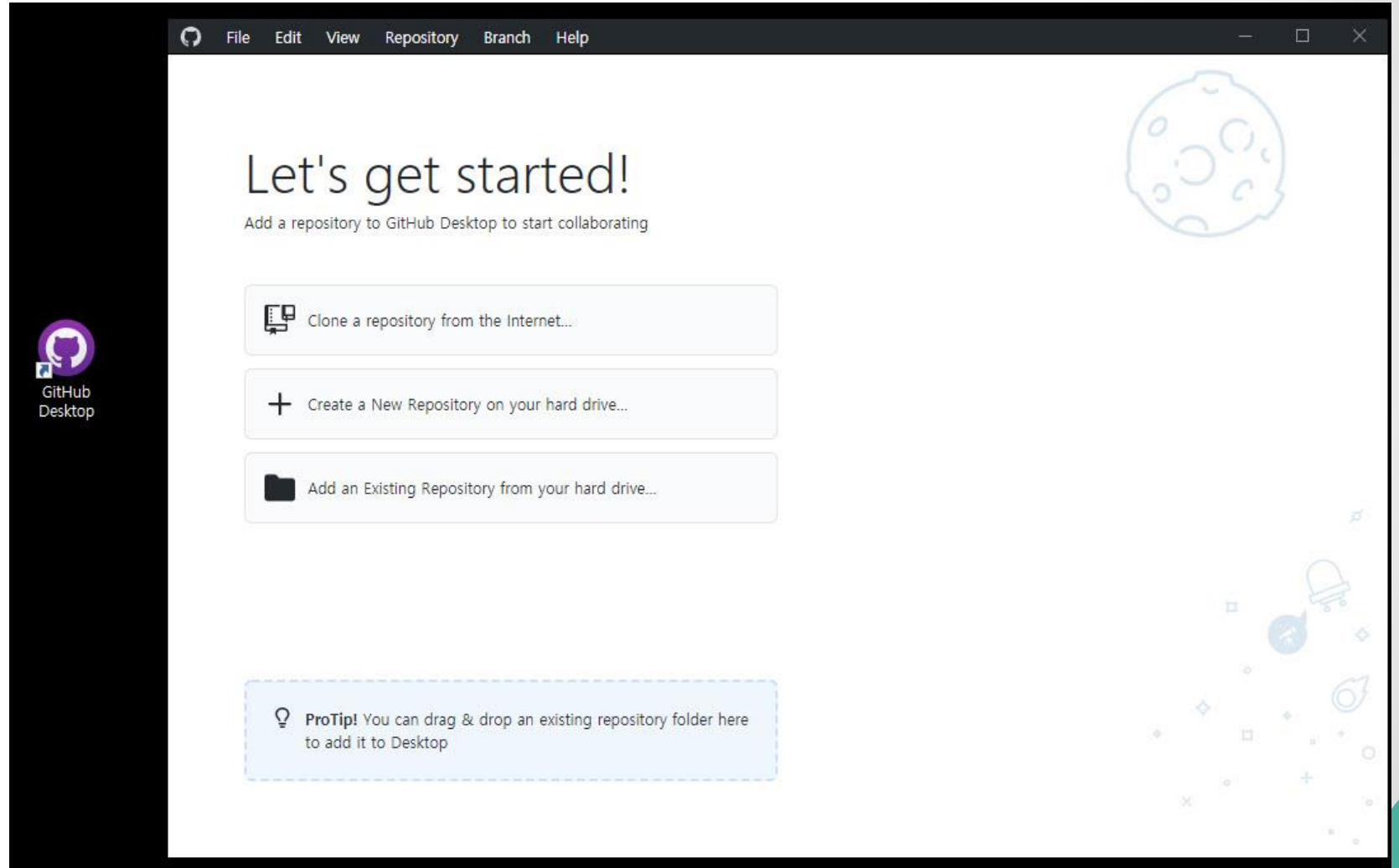
버전 관리

- 데스크톱에 깃허브 리포지토리 복사하고(clone), 새로운 유니티 프로젝트 생성하기
- 아래 웹사이트에서 깃허브 데스크톱(github desktop)을 다운로드 및 설치한다.
- <https://desktop.github.com>



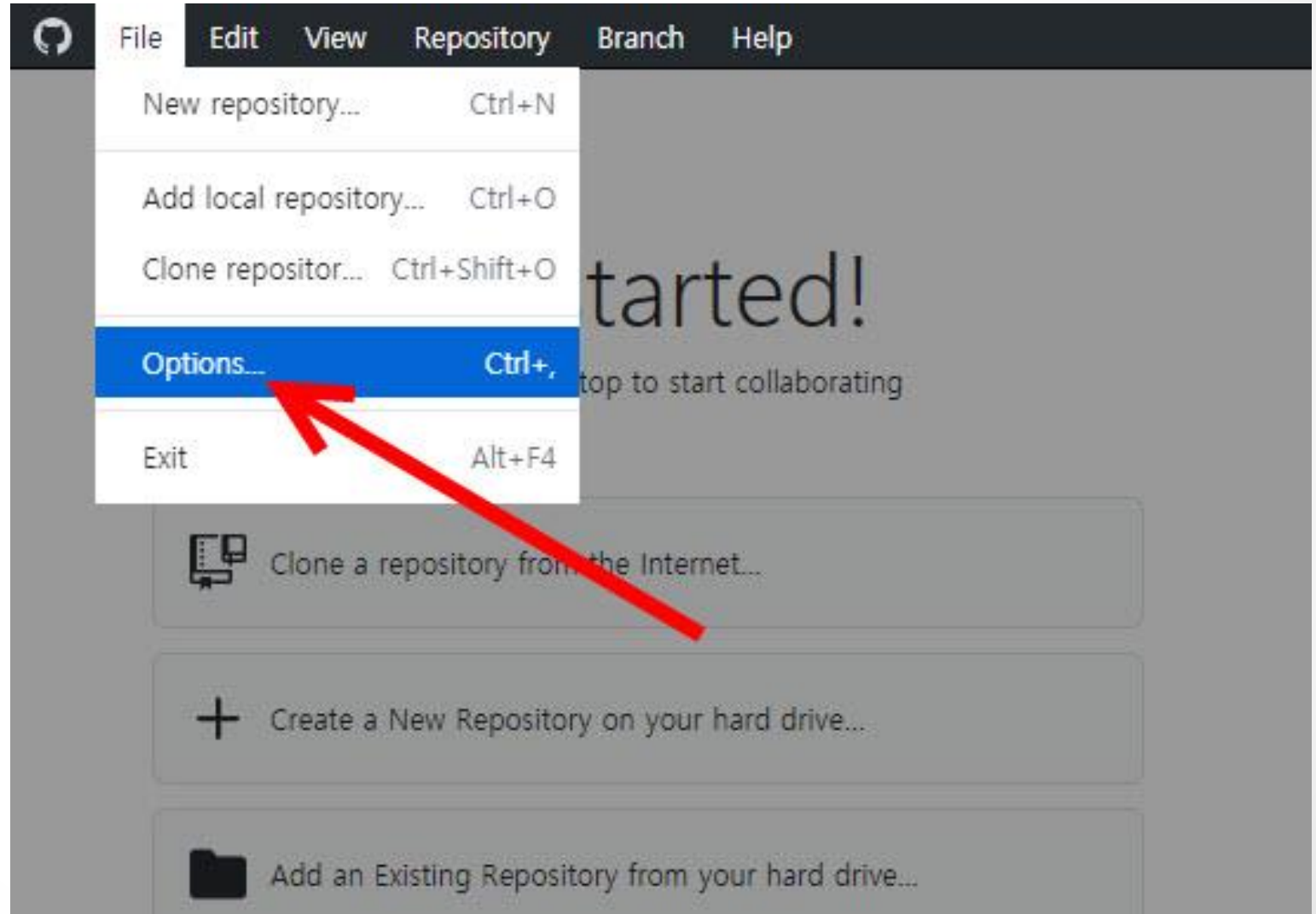
버전 관리

- 깃 허브 데스크톱을 실행!



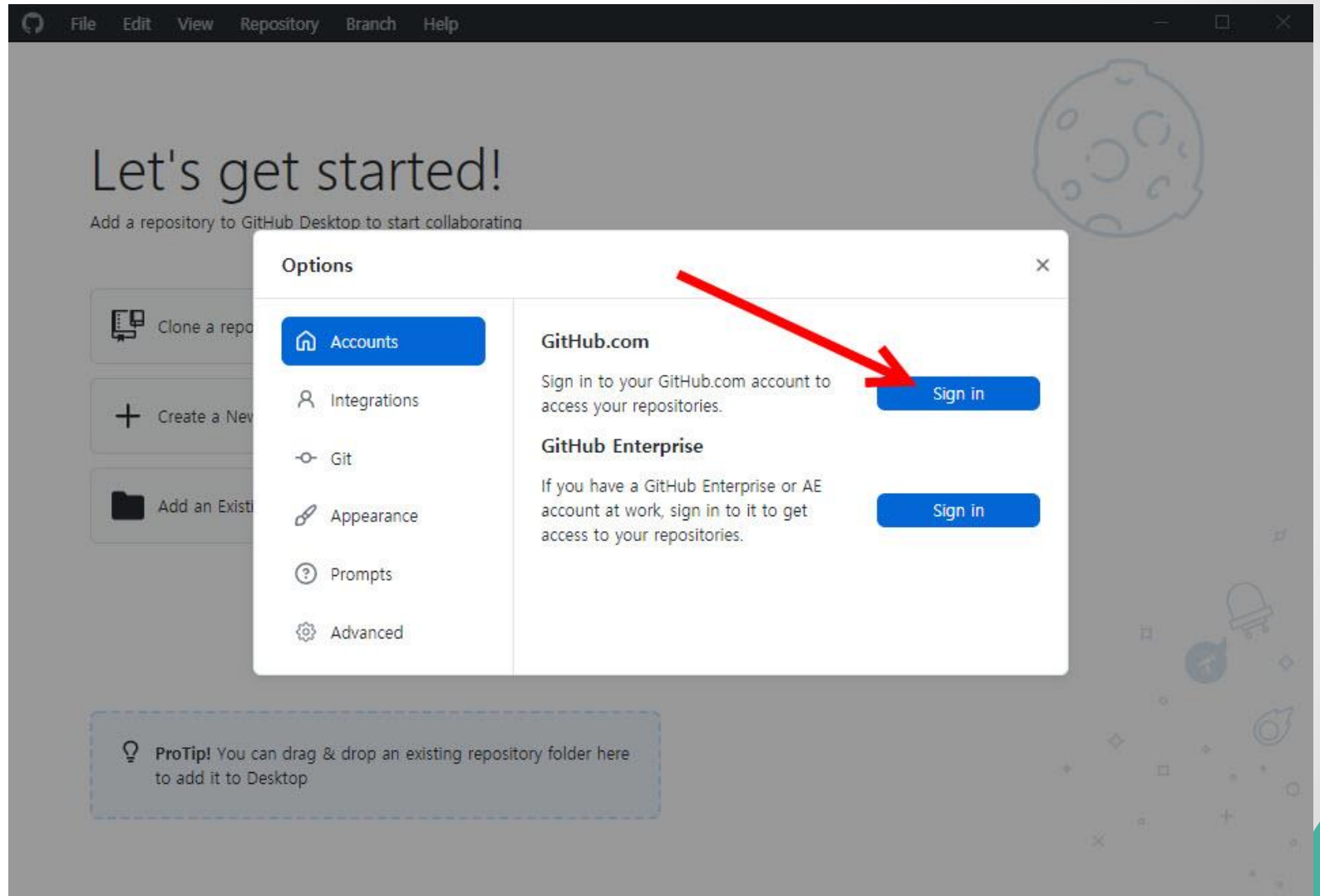
버전 관리

- 상단 file 메뉴의 options으로 이동한다.



버전 관리

- accounts에서 깃허브의 계정으로 로그인하고 데스크톱의 접근 권한을 부여한다.





Authorize GitHub Desktop



GitHub Desktop by **desktop**
wants to access your **hcijun** account



Repositories
Public and **private** ▼



Personal user data
Full access ▼



Workflow
Update GitHub Action Workflow files. ▼

Cancel

Authorize desktop

Authorizing will redirect to
`x-github-desktop-auth://oauth`



Owned & operated
by GitHub



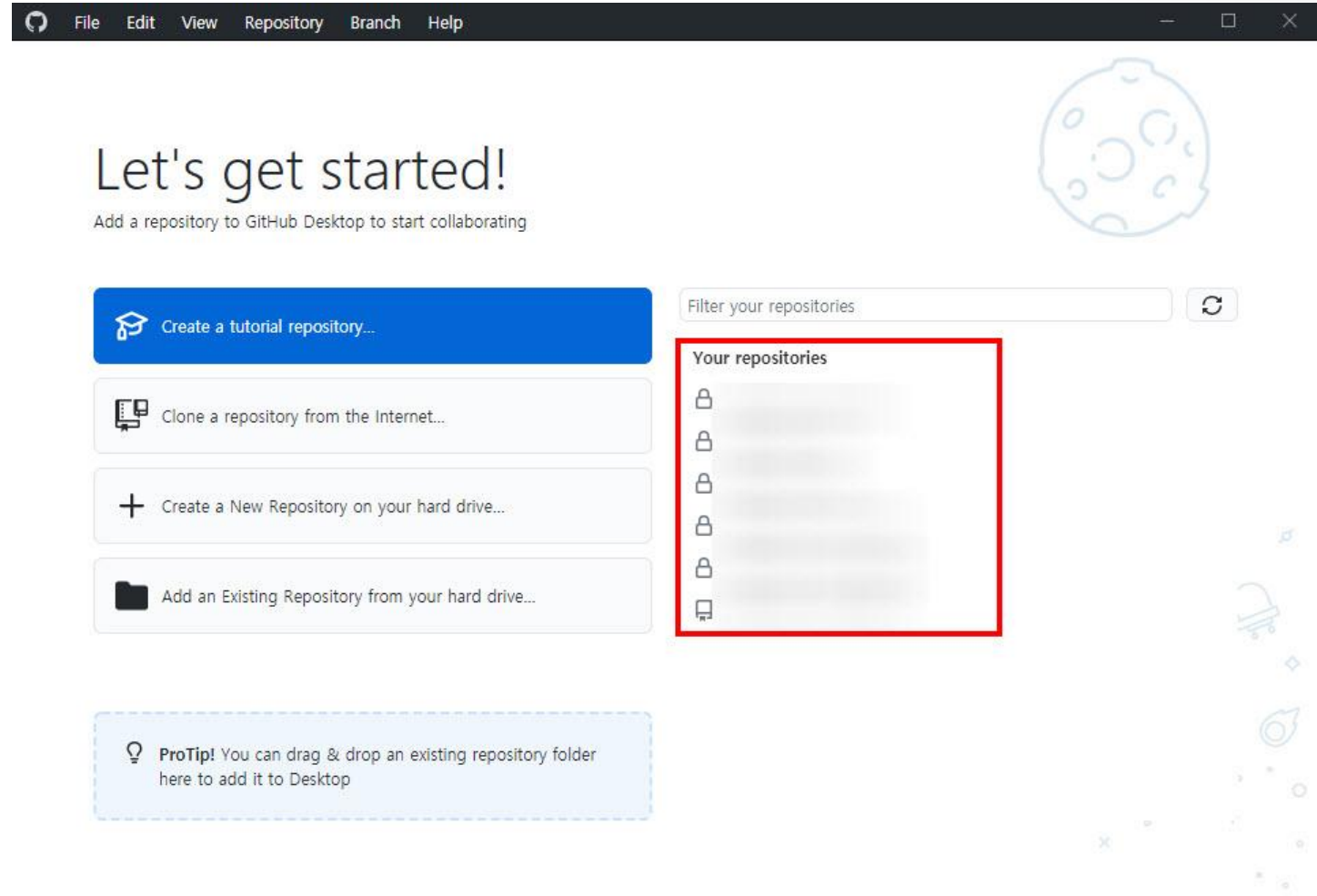
Created 5 years ago



More than 1K
GitHub users

버전 관리

- 계정 로그인 완료되면 깃허브의 리포지토리들을 표시한다.



버전 관리

- clone a repository from internet을 선택

Let's get started!

Add a repository to GitHub Desktop to start collaborating



Create a tutorial repository...



Clone a repository from the Internet...



Create a New Repository on your hard drive...



Add an Existing Repository from your hard drive...



ProTip! You can drag & drop an existing repository folder here to add it to Desktop

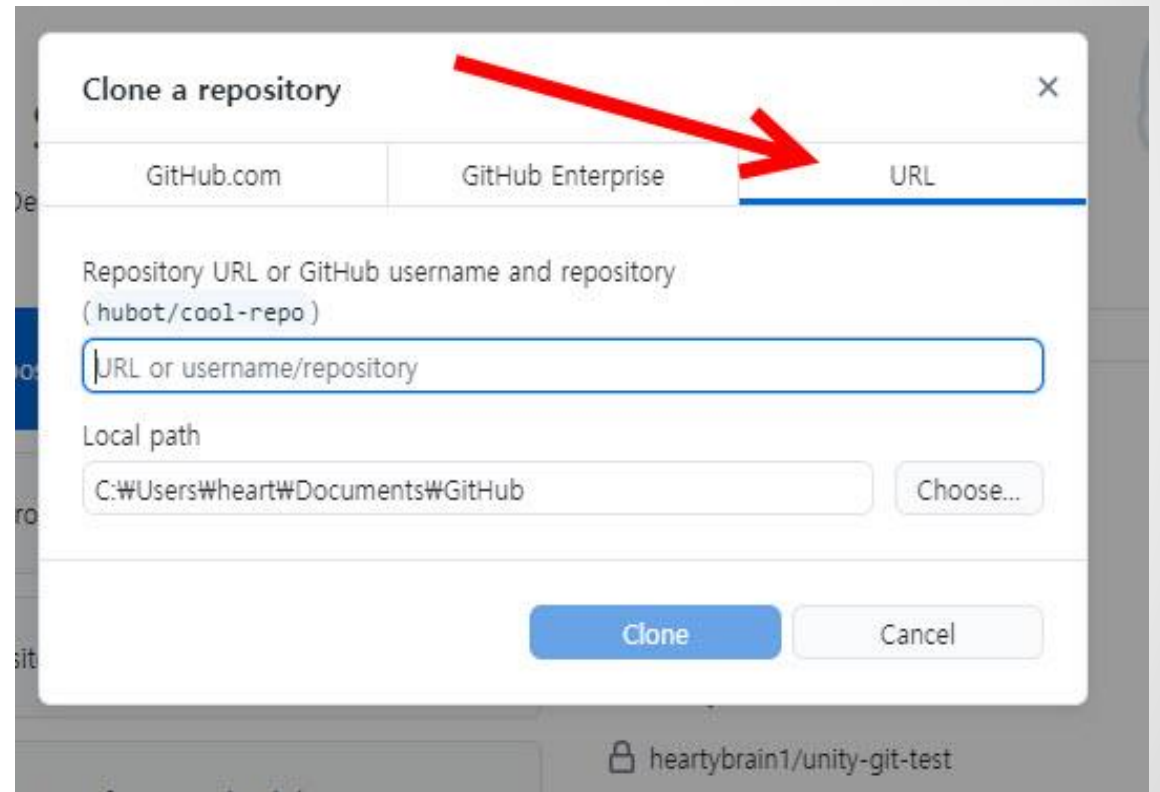
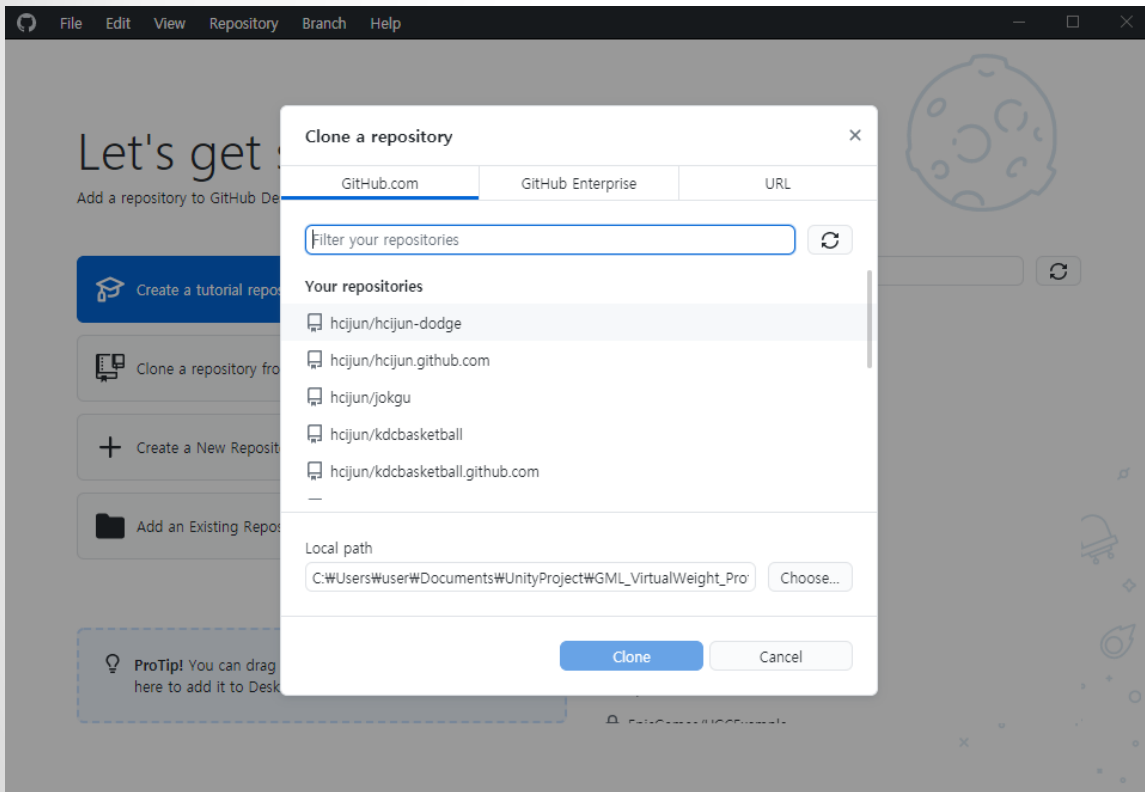
Filter your repositories

Your repositories



버전 관리

- clone a repository에서 복사 방법을 선택
- 내가 만든 프로젝트 혹은 이미 가입한 프로젝트라면 바로 GitHub.com 에서 선택 가능
- 일반적인 방법인 URL을 사용하여 리포지토리를 복사하려면 URL탭으로 이동



버전 관리

- 다음을 클릭하여 주소를 복사하기!

hcijun / hcijun-dodge

<> Code ⓘ Issues 🔗 Pull requests ⌚ Actions 📁 Projects 📖 Wiki 🛡 Security 📈 Insights ⚙ Settings

main 1 branch 0 tags

Go to file

Add file ▾

Code ▾

hcijun Update README.md

.gitignore

Initial commit

README.md

Update README.md

README.md

hcijun-dodge

닷지 플레이어 게임 입니다. 재미있는 게임이 되고 싶습니다

Clone ⓘ

HTTPS SSH GitHub CLI **New**

https://github.com/hcijun/hcijun-dodge.git



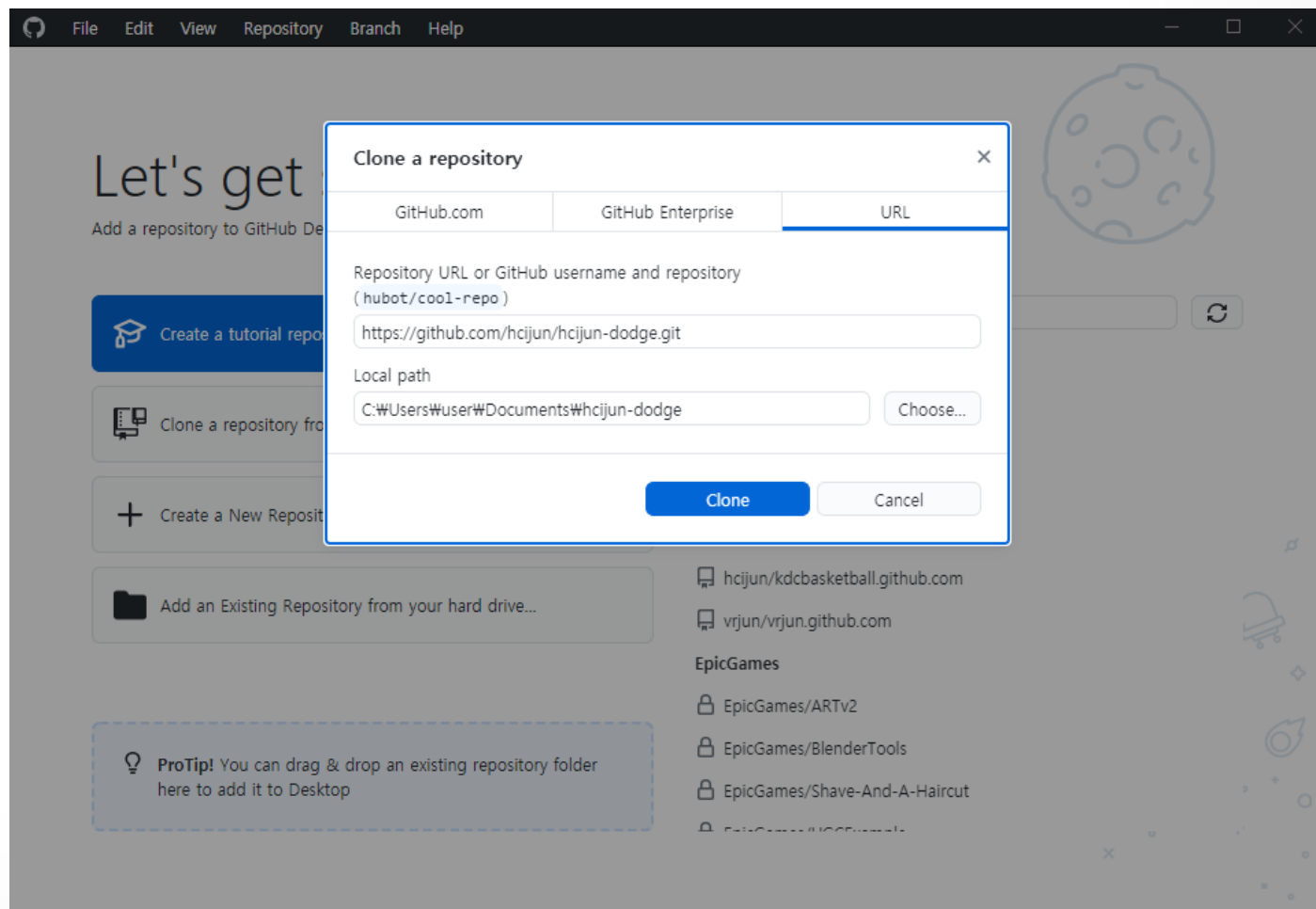
Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

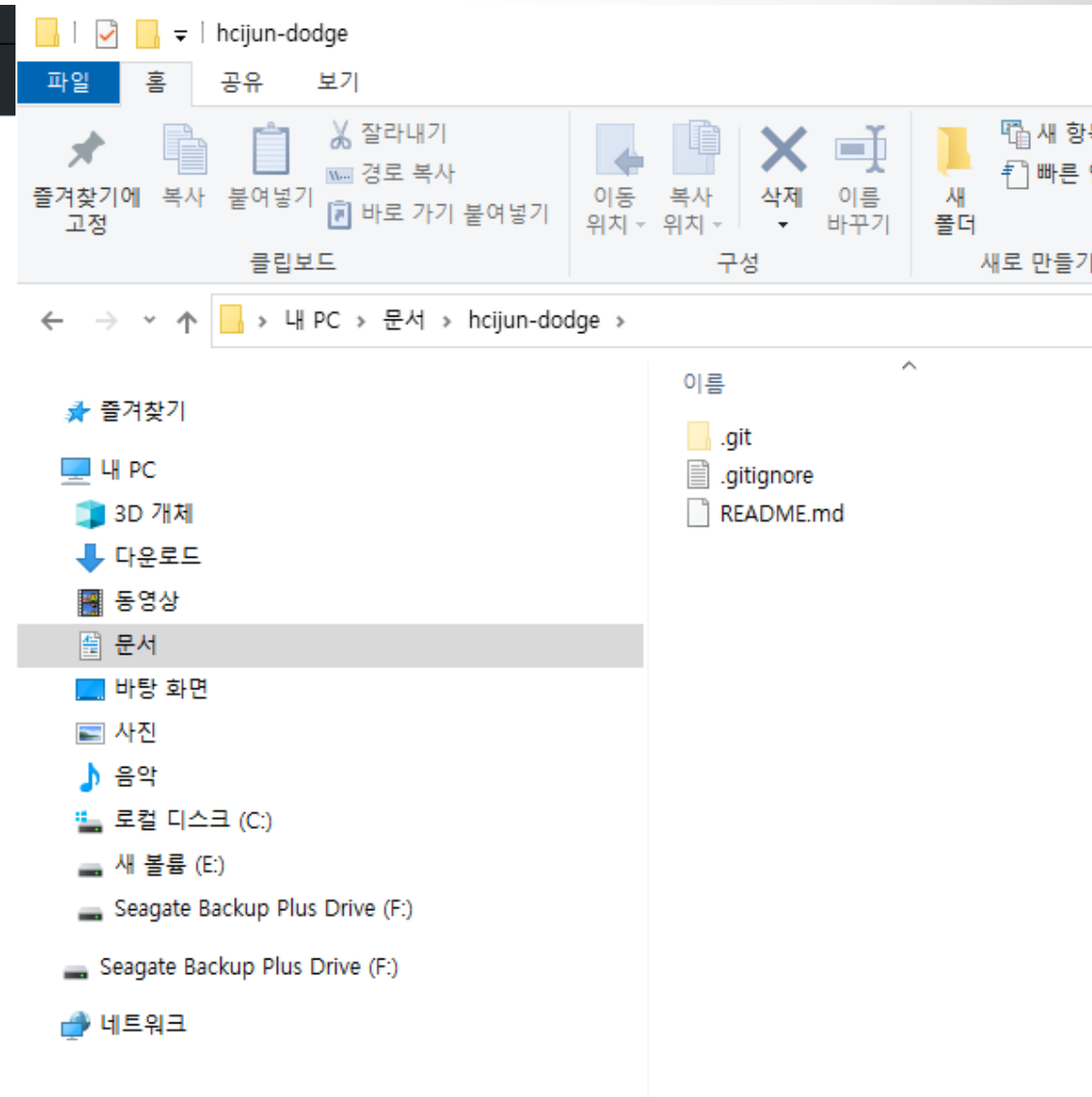
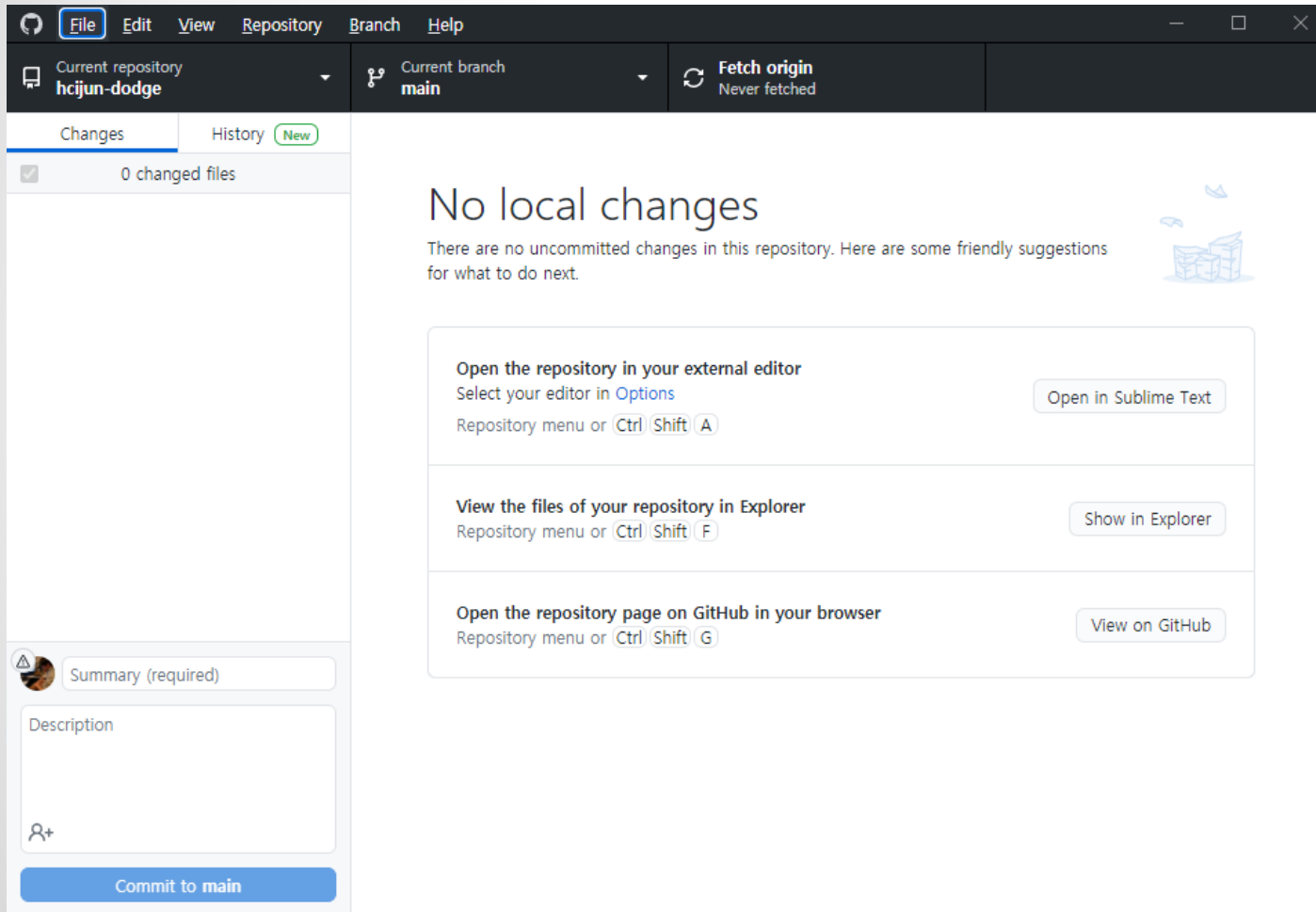
버전 관리

- URL에 주소를 복사 및 경로를 설정하고 Clone 클릭!



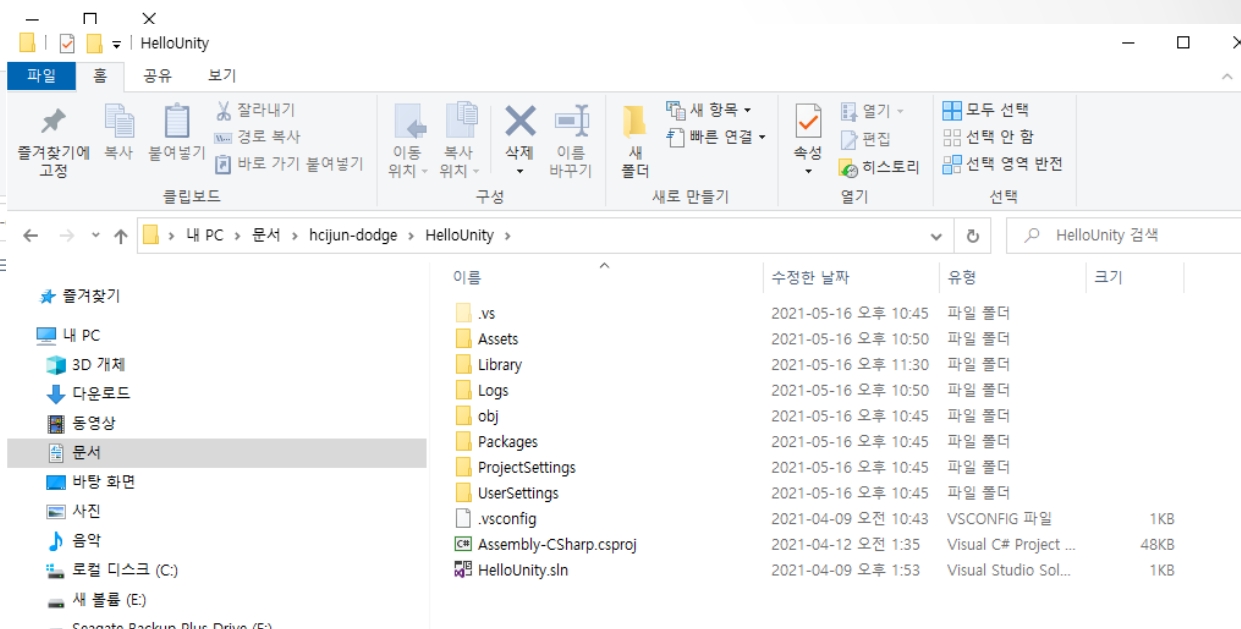
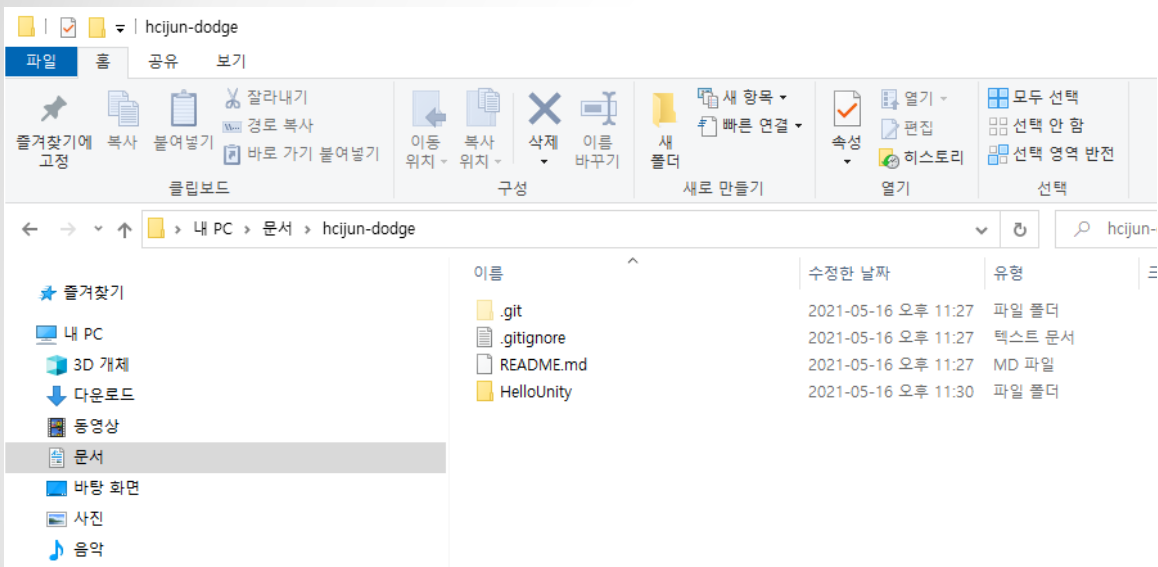
버전 관리

- 다음과 같은 화면이 보임



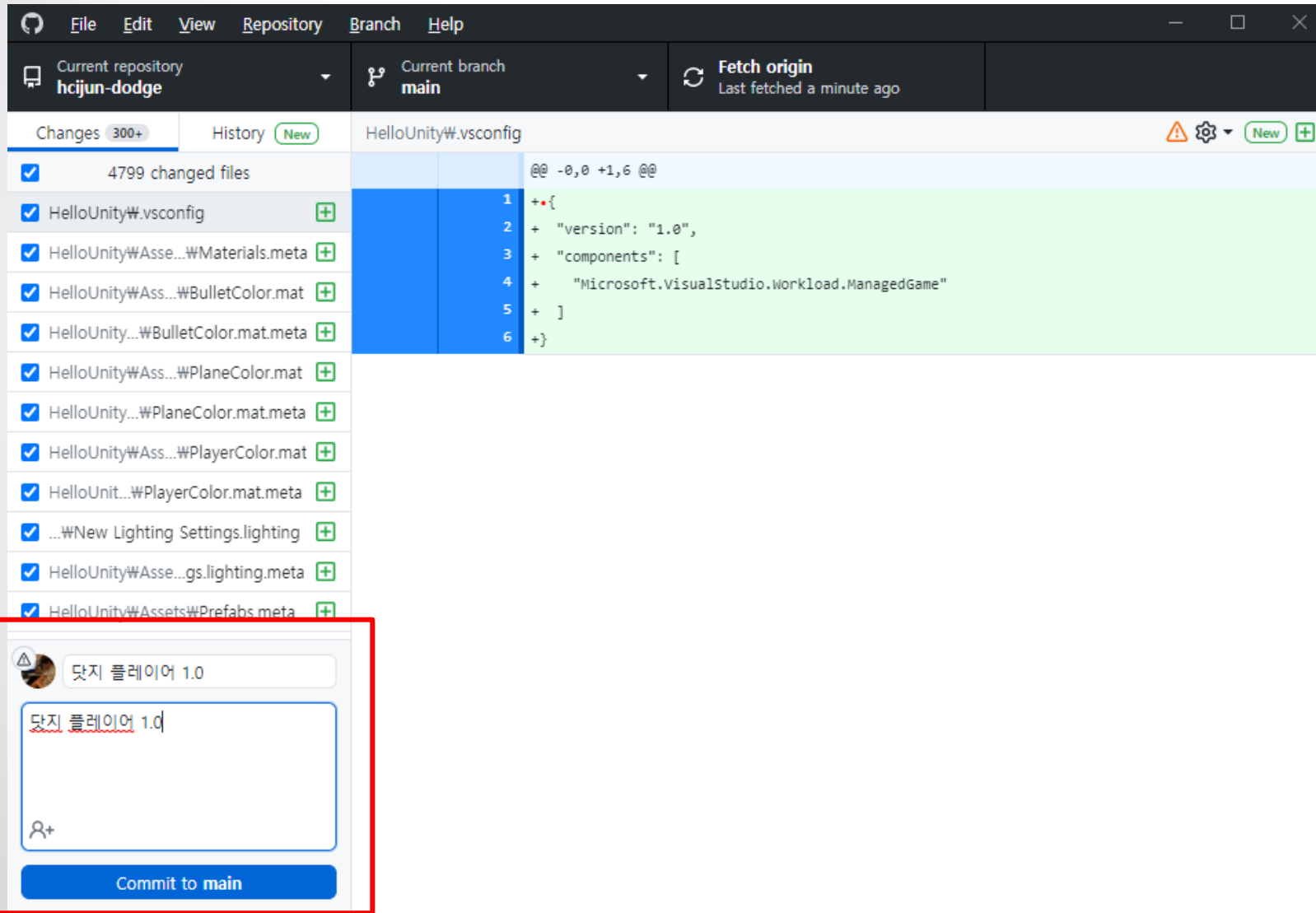
버전 관리

- 이 리포지토리에 우리가 만든 유니티 프로젝트를 추가해 보자



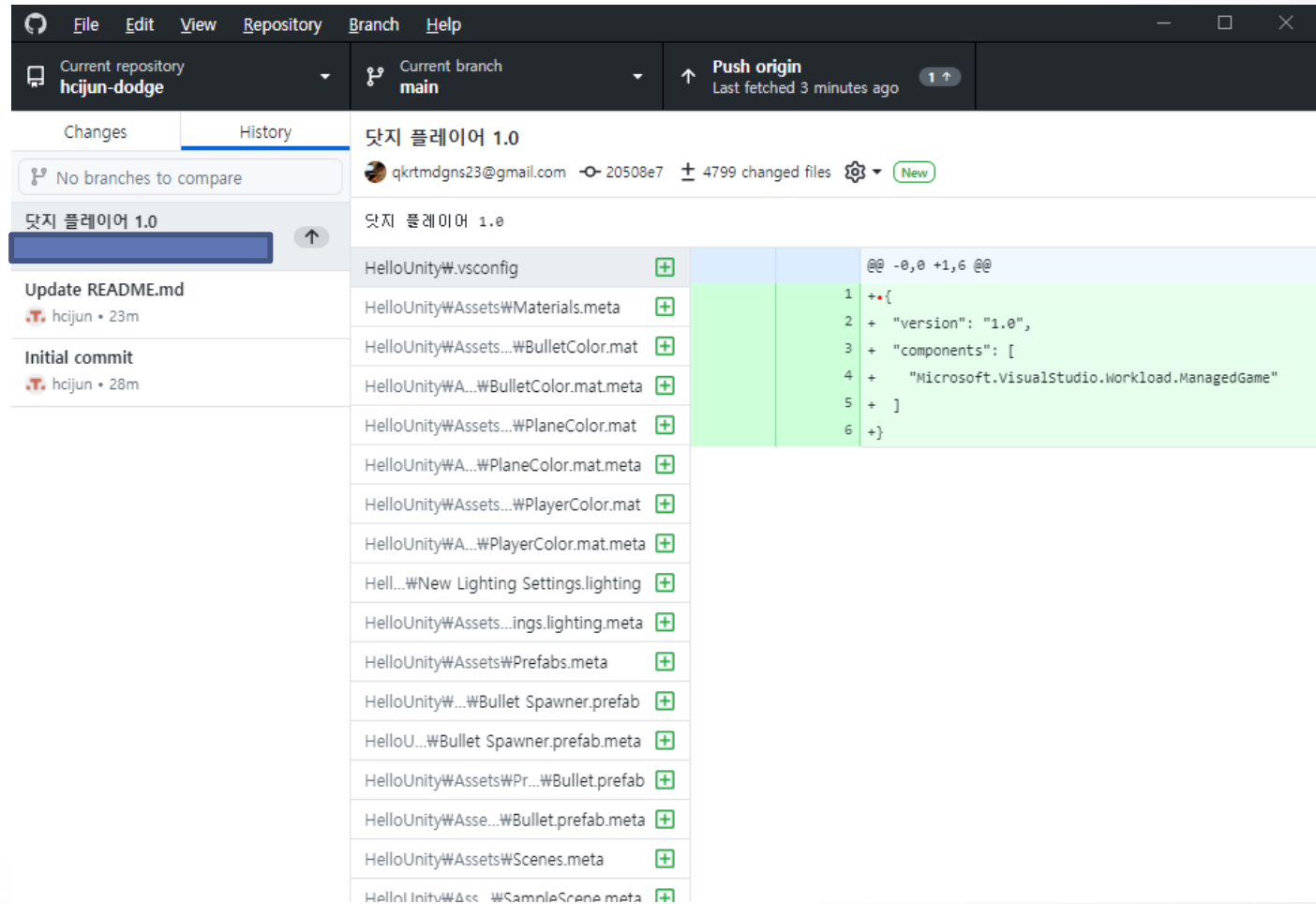
버전 관리

- 깃허브 데스크톱에서 리포지토리 변경사항을 표시한다. 유니티 프로젝트에서 생성한 파일 및 폴더의 추가사항을 확인할 수 있음!



버전 관리

- 닷지 플레이어가 로컬의 저장소에 commit 가 됨!
- 이것을 깃 허브에있는 원격 저장소에 업로드를 하기 위해서는 Push 를 실행 해야함



버전 관리

- 이제 웹사이트에서 업데이트가 된 것을 볼 수 있음!

hcijun / hcijun-dodge

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main hcijun-dodge / HelloUnity / Go to file Add file ...

...

20508e7 3 minutes ago History

Assets	닷지 플레이어 1.0	3 minutes ago
Library	닷지 플레이어 1.0	3 minutes ago
Logs	닷지 플레이어 1.0	3 minutes ago
Packages	닷지 플레이어 1.0	3 minutes ago
ProjectSettings	닷지 플레이어 1.0	3 minutes ago
UserSettings	닷지 플레이어 1.0	3 minutes ago
obj/Debug	닷지 플레이어 1.0	3 minutes ago
.vsconfig	닷지 플레이어 1.0	3 minutes ago