


[GitHub]초보자도 할 수 있는 깃허브 사용법

호IT · 2020. 10. 21. 03:15



실전 연애 활용 100%! 연애유튜버 미란다TV만의 '데이팅 가이드북'




- GitHub?

처음 들어보신 분들도 있고, 들어는 봤지만 너무 복잡해 보여서 시작하기 힘들었던 분들도 계실 텐데요.
깃허브는 소프트웨어 개발 프로젝트를 목적으로 한 코드 관리 서비스입니다!
크게 공들이지 않고 자신이 했던 증거를 남긴다고 생각하면 될 것 같은데,
많은 프로그래머분들이 포트폴리오용으로 쓰고 계십니다.

- Git 설치

<https://gitforwindows.org/>



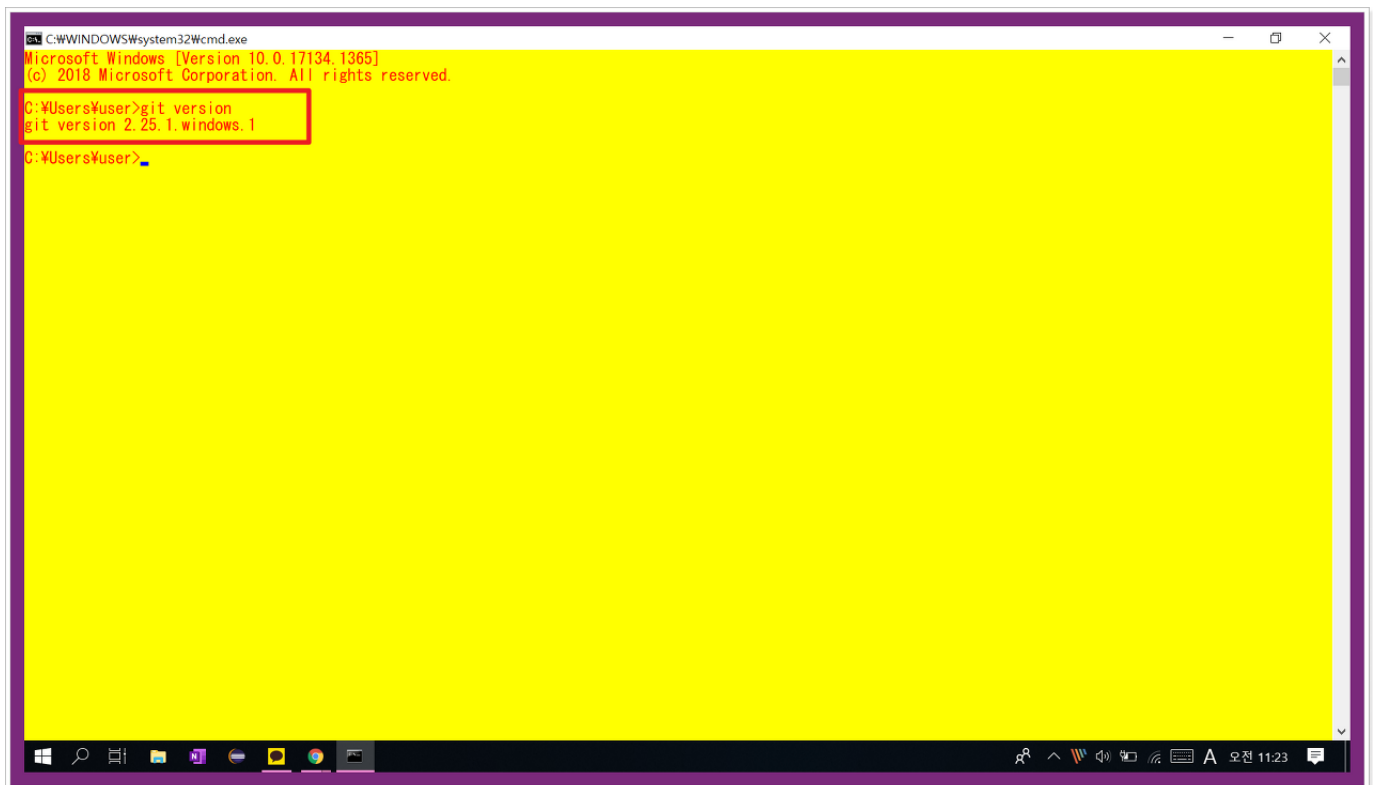
Git for Windows

gitforwindows.org

들어가서 깃을 다운로드해줍니다.

저는 기본 옵션으로 다 next 해주면서 설치했어요.

설치 확인은 cmd 창을 열고 (윈도 키+ R)



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.17134.1365]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\User>git version
git version 2.25.1.windows.1

C:\Users\User>
```

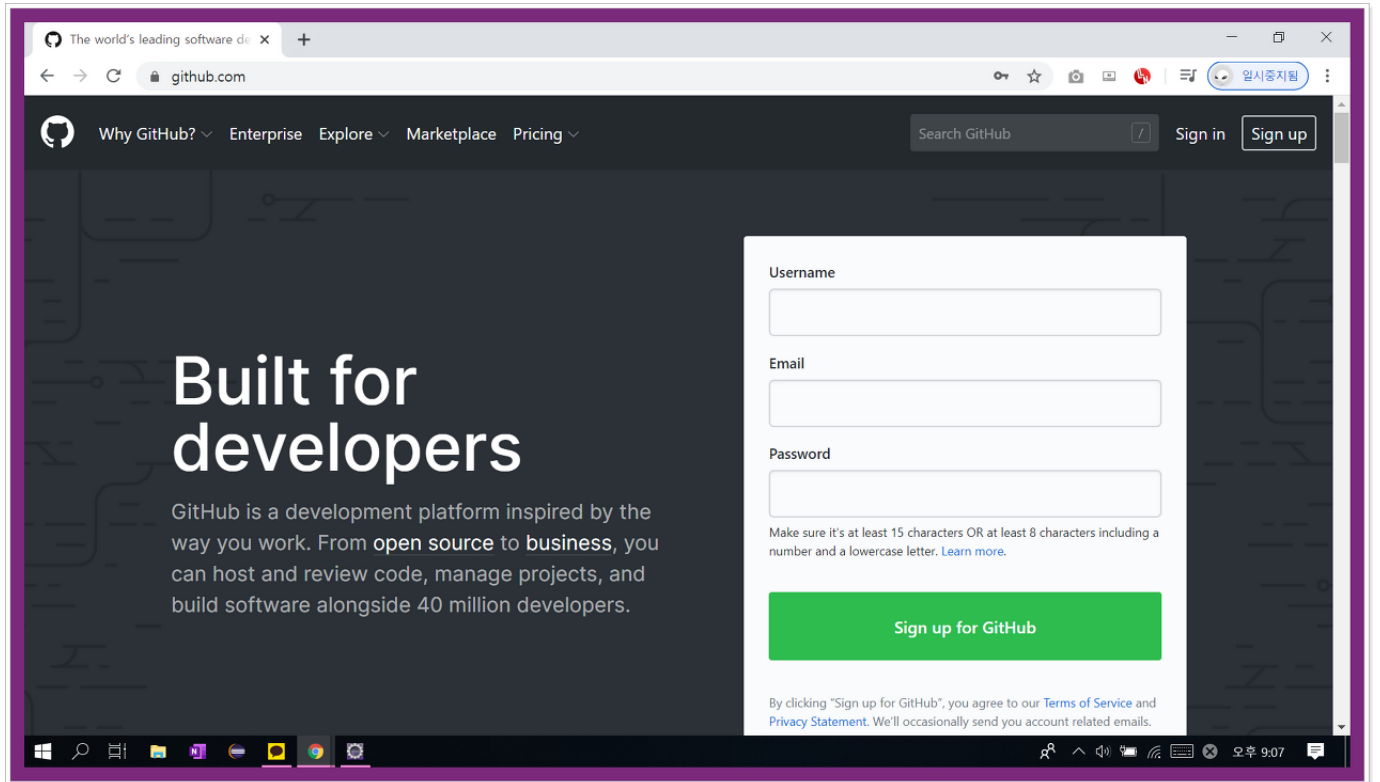
깃 버전이 뜨면 설치가 완료

- GitHub Sign Up

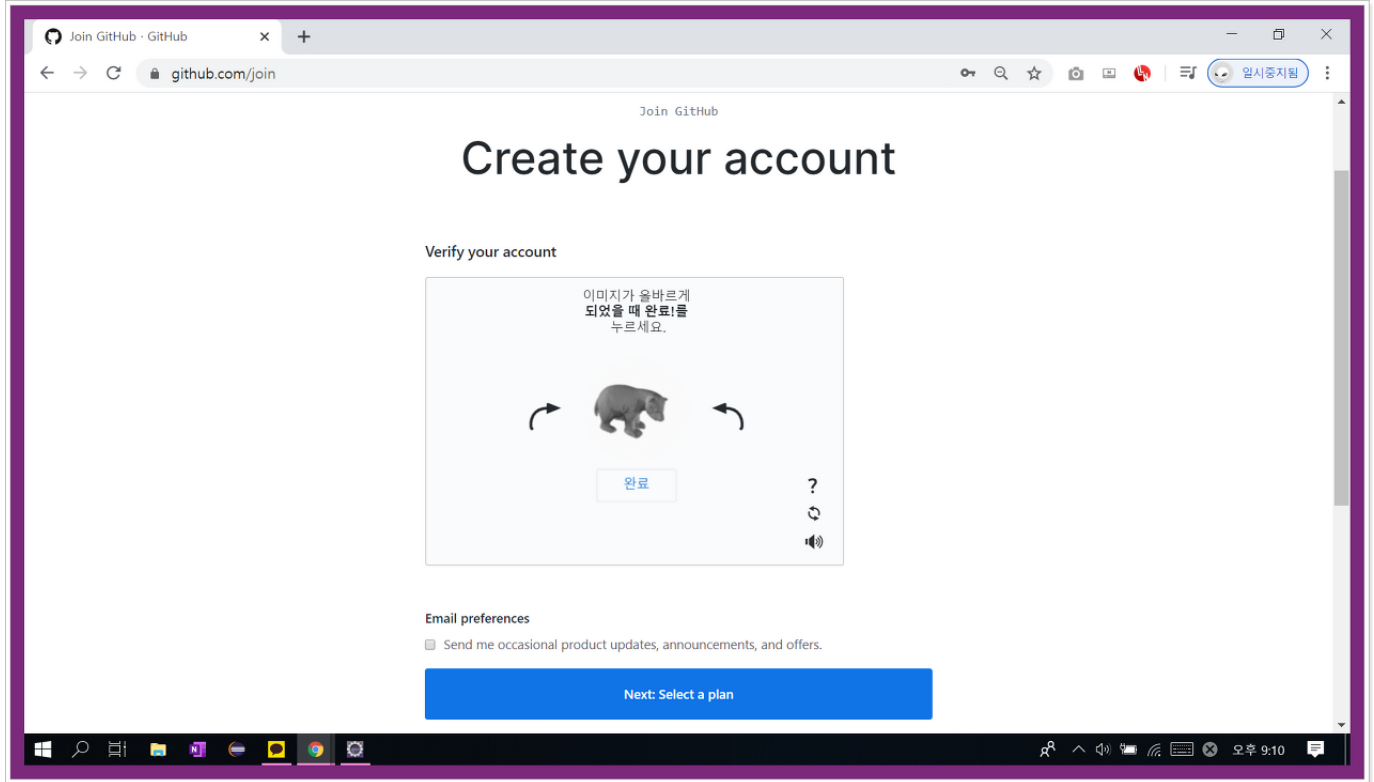
우선 가입을 해볼게요.

<https://github.com/>

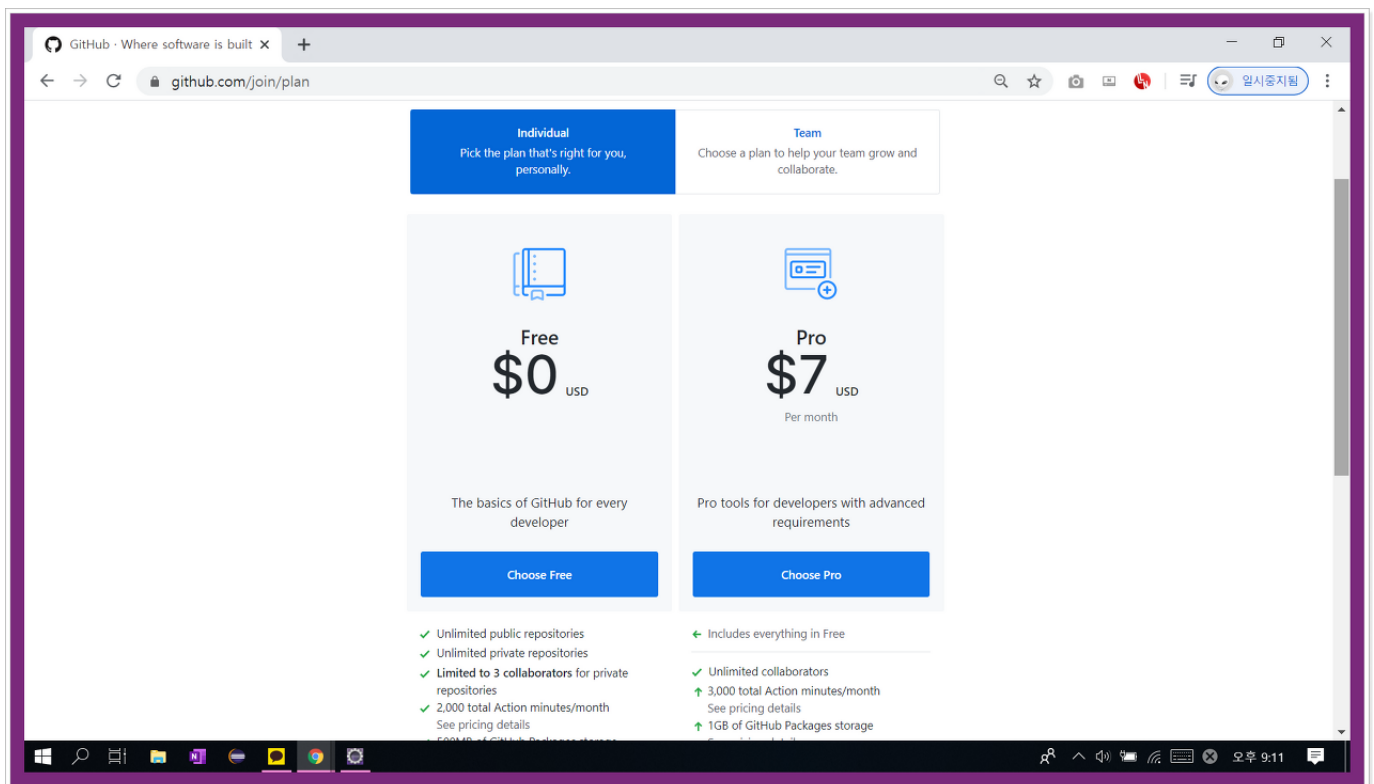
위의 사이트에 들어가서



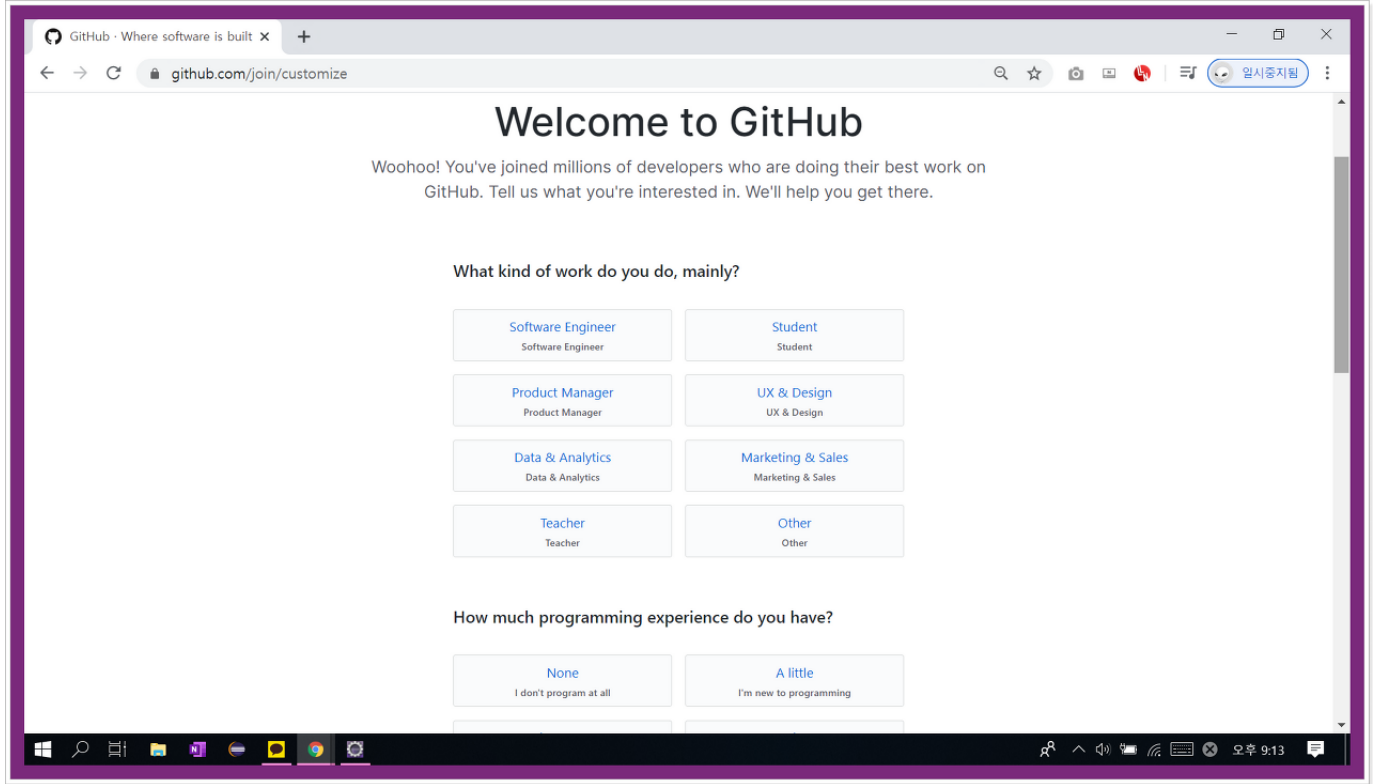
Username, e-mail, Password 칸을 채우시고 SignUp for Github 버튼을 누릅니다.



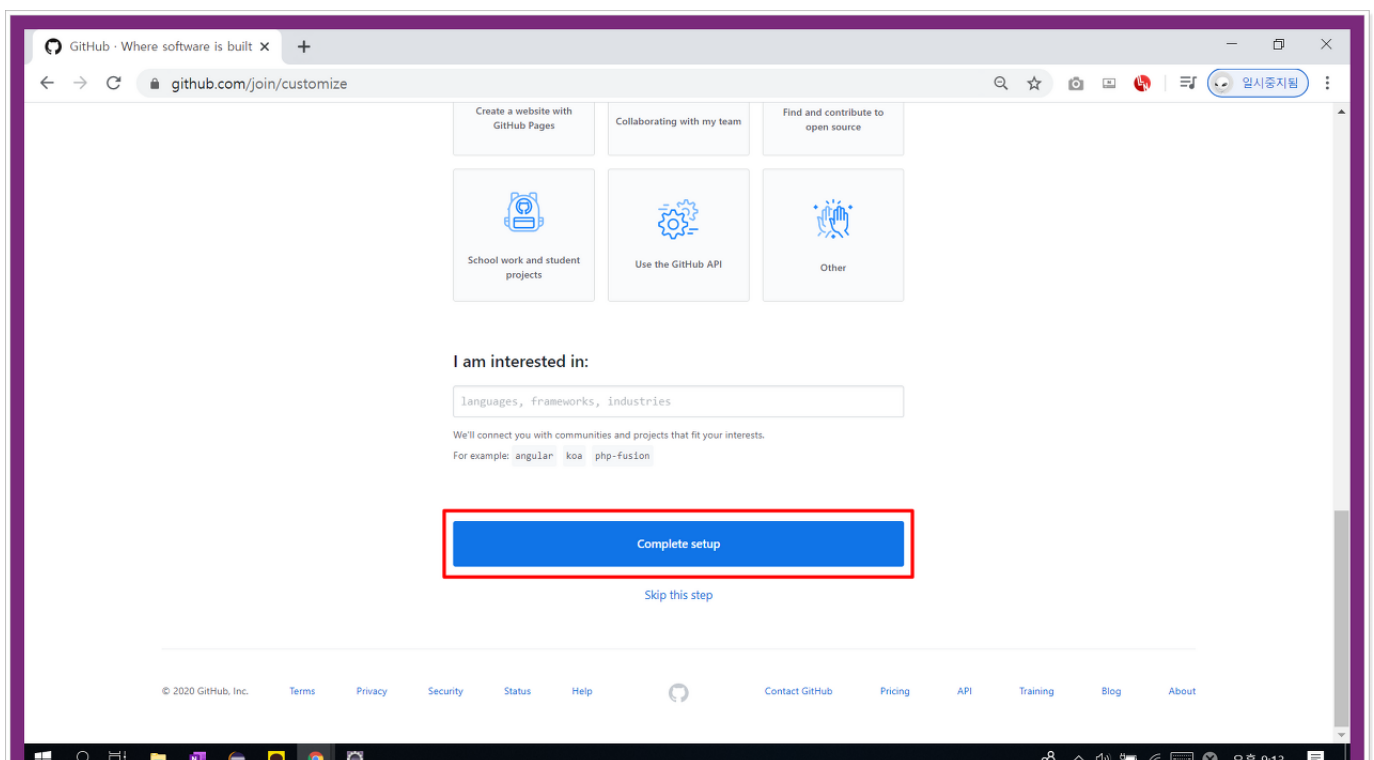
실제 사용자의 계정인지 간단하게 인증하고, 넘어갑니다.



우리는 개인, 그리고 무료 버전을 사용할 것이기 때문에 Choose Free 버튼을 클릭



어떠한 용도로 깃허브를 사용할 것인지 간단한 설문조사



Complete Setup을 눌러주면 자신의 메일 계정에 깃허브가 보낸 메일이 옵니다. 인증해주면 가입은 완료

- GitHub사용 방법

우선 몇 가지 용어들을 정리해봅시다.

- 커맨드 라인 (Command Line)

윈도에서는 명령 프롬프트(cmd)에서, 맥이나 다른 OS에서는 터미널이라고 하는 프로그램입니다.

우리가 일반적으로 접하는 마우스로 클릭하는 프로그램이 아닌 키보드로 입력을 받아 명령을 수행합니다.

- 저장소 (Repository)

프로젝트가 살고 있는 저장공간. (혹은 디렉토리라고 칭하기도 합니다.)

다른 블로그나 카페글들을 보니 주로 'repo'라고 줄여서 사용하고 있네요!

이 저장소 안에 코드, 텍스트, 이미지를 저장할 수 있습니다.

- 버전 관리 (Version Control)

팀 단위로 프로젝트를 운영할 때 필수적인 기능인데, 깃허브는

다수의 사용자가 하나의 페이지를 수정해도 각 사용자별로 수정 내용을 저장합니다.

모든 시점의 '스냅 샷'을 저장하기 때문에 겹쳐 쓸 일이 없습니다.

- 커밋 (Commit)

보통의 체크포인트가 그렇듯이 수정을 하다 이 체크포인트로 되돌아가고, 복원할 수 있습니다.

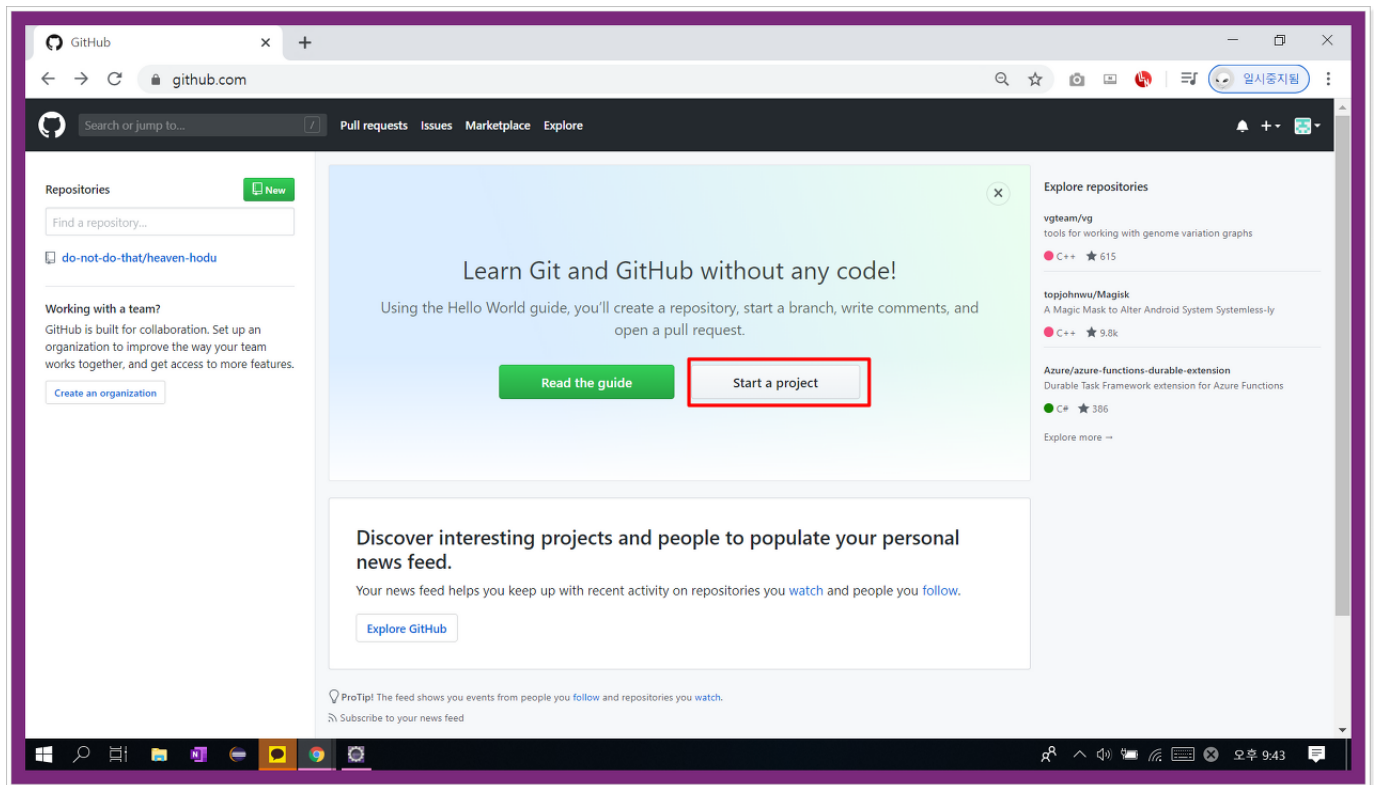
- 브랜치 (Branch)

자신이 진행하는 프로젝트의 브랜치를 따서 (Branch off) 자기의 버전을 만드는 겁니다.

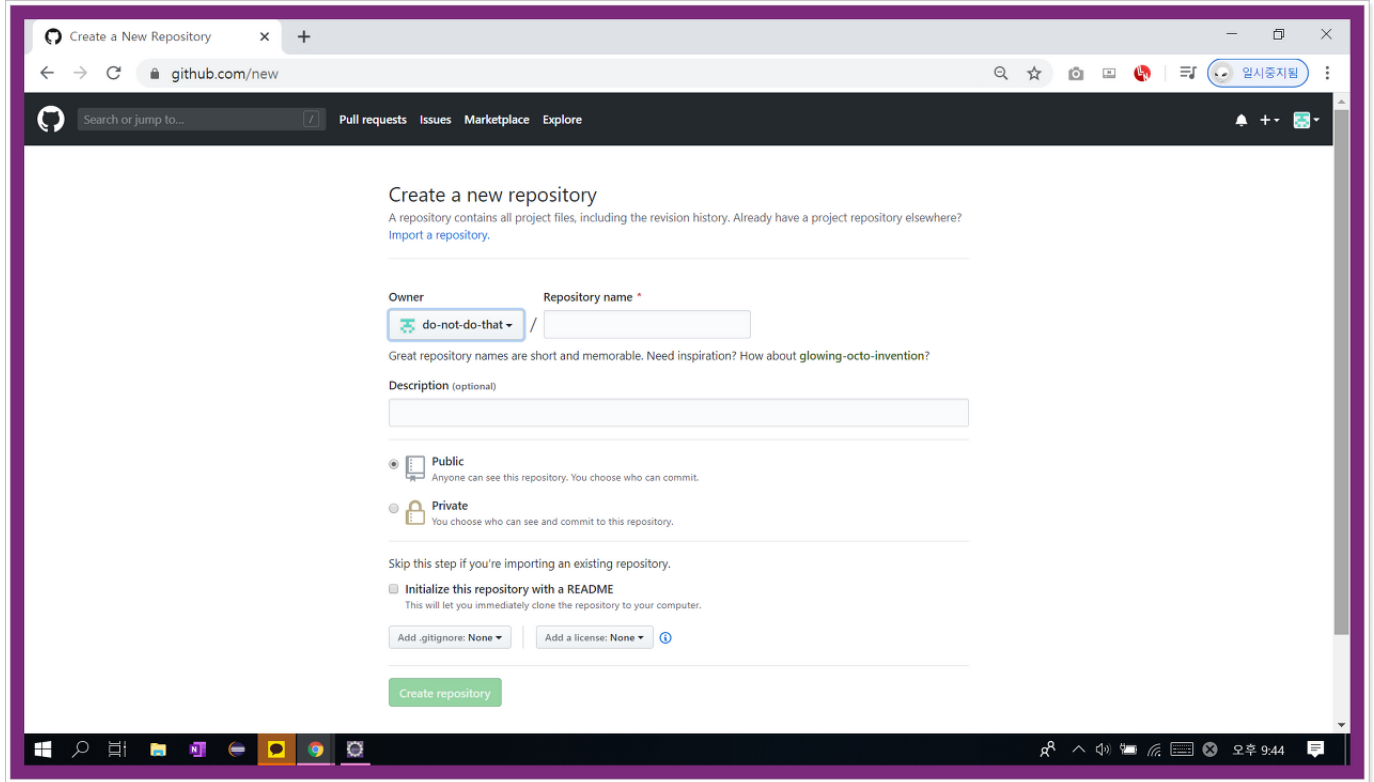
작업을 마무리하면 프로젝트 메인디렉토리인 'master'에 브랜치를 'merge'합니다.

기본적인 용어를 정리해보았는데, 이제 본격적으로 사용방법을 알려드릴게요.

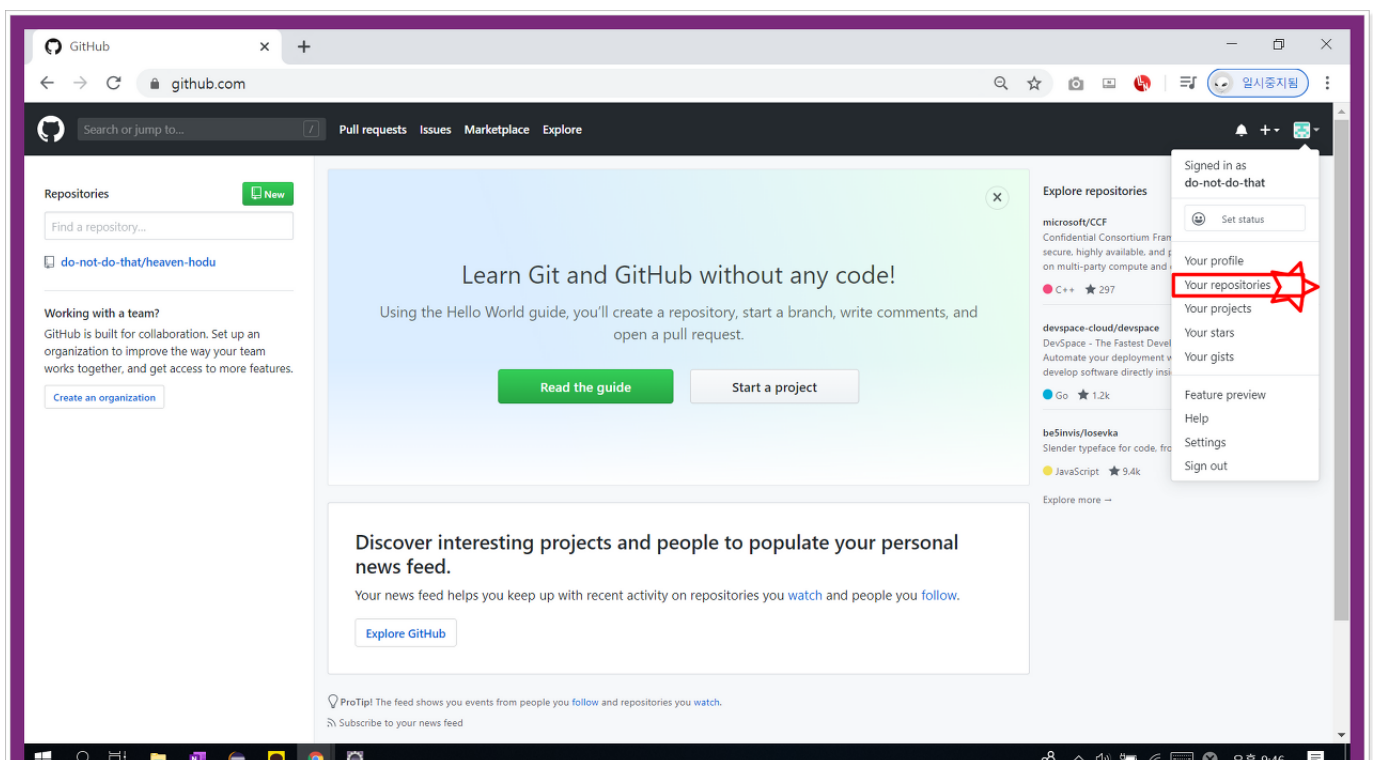
우선 저장소를 만들어보겠습니다.



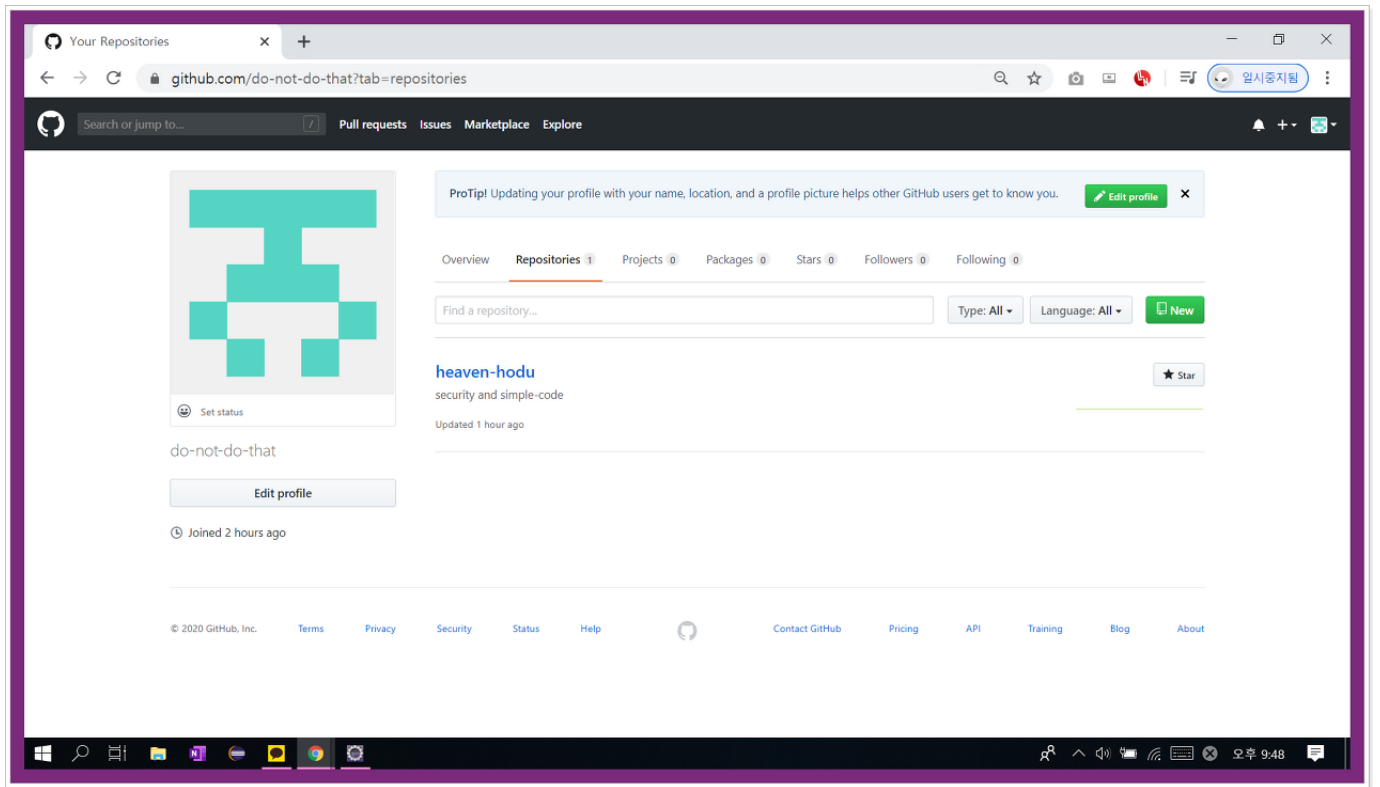
로그인을 하고, Start a project를 클릭해줍니다.



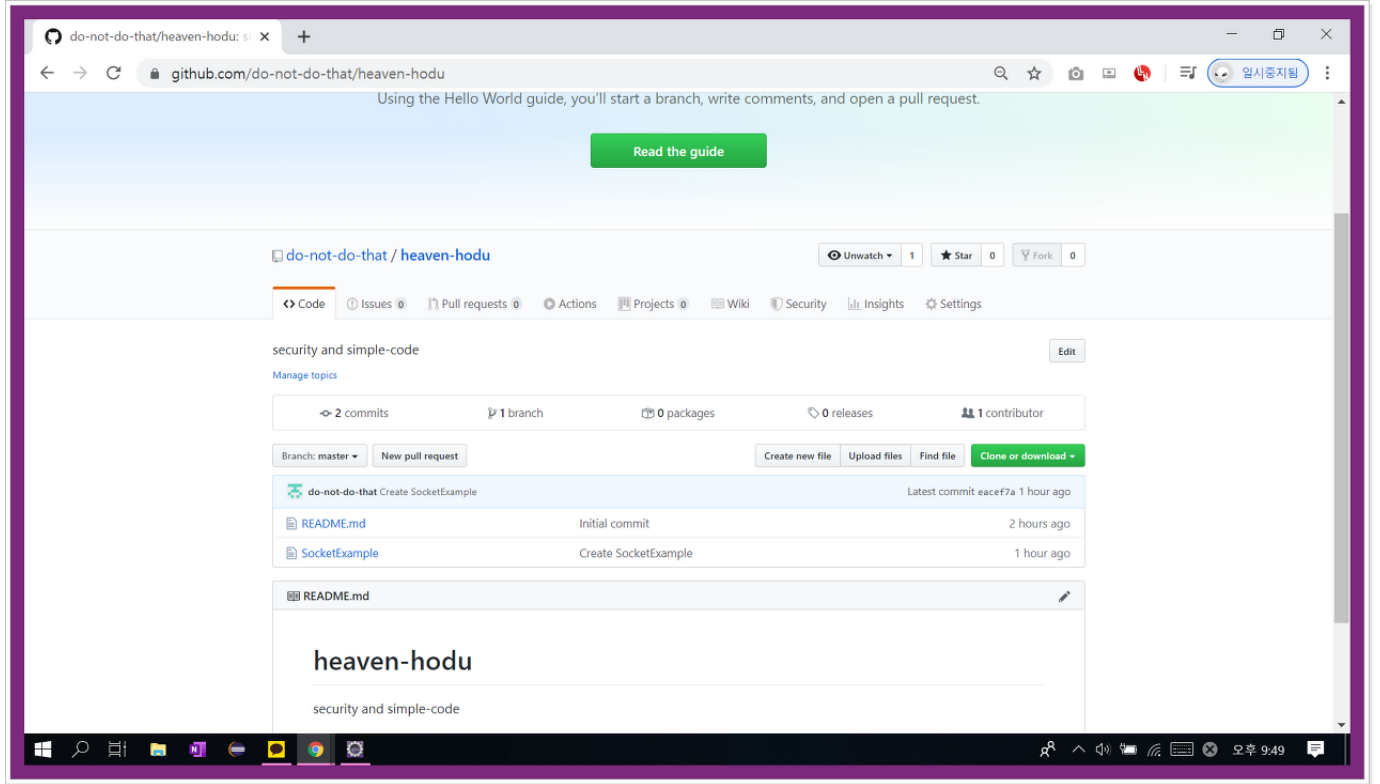
Repo의 이름과 주석 등을 작성하고, Create repository를 눌러서 저장소를 하나 만듭니다.



만들어진 repo는 자신의 프로필 클릭하고 Your Repositories 누르면 들어갈 수 있어요!



그럼 이렇게 방금 만든 저장소가 보이고, 이를 클릭하면!



저장소 안에 있는 파일이 보입니다.

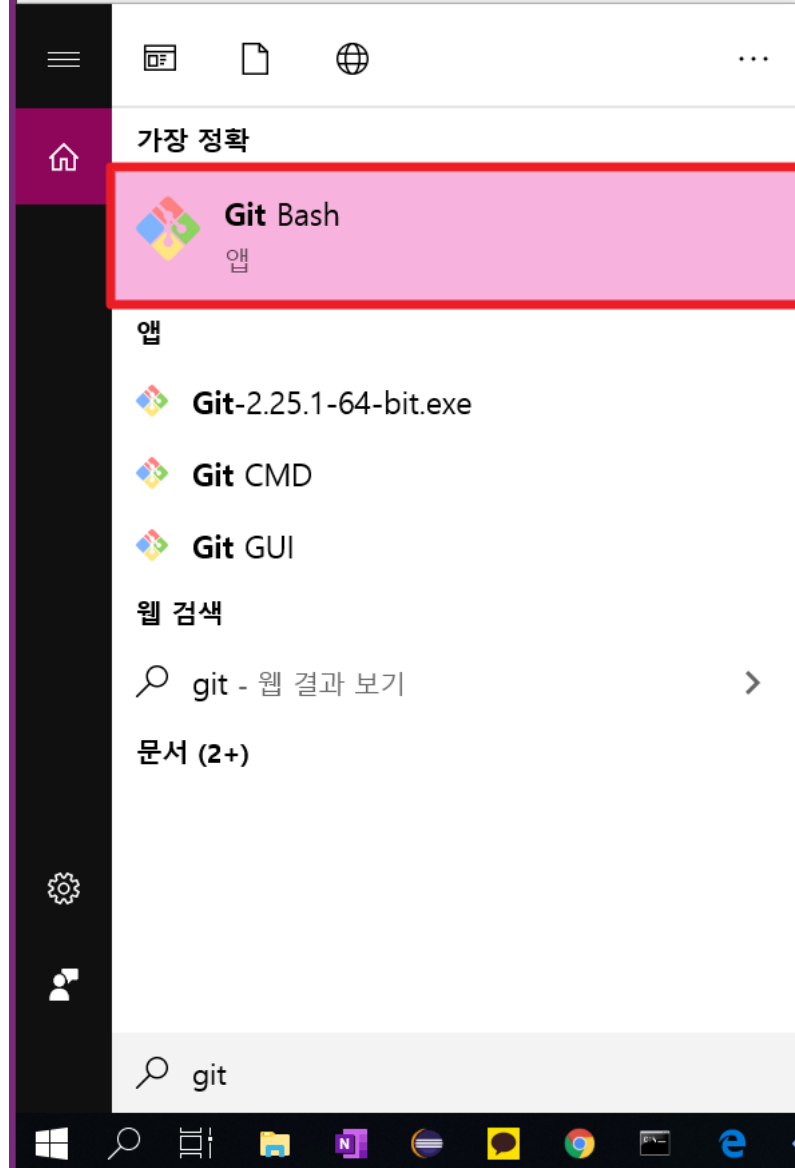
저는 SocketExample이라는 파일을 넣었는데, 원래는 README.md라는 파일만 있을 거예요.

자신의 Git Repo 주소는 [https://github.com/\(아이디\)/\(repo 이름\). git](https://github.com/(아이디)/(repo 이름). git)입니다.

저의 Git Repo 주소는 <https://github.com/do-not-do-that/heaven-hodu> 가 되겠네요.

물론 화면에 보이는 Clone or download라는 초록색 버튼을 눌러도 자신의 깃 레포지토리 주소가 보입니다.

이 주소를 이용해서 앞으로 소스코드를 수정하고, 올리고 받고 할 겁니다.



깃 배쉬를 이용해봅시다.

돋보기 버튼을 눌러서 git bash를 검색해 들어가 줍니다.

먼저 여기서 우리가 해야 할 작업이 있습니다.

```
git config --global user.name "유저 이름"
```

```
git config --global user.email "가입할 때 사용한 메일 주소"
```

를 작성해줍니다.

```
MINGW64/c/Users/user/Desktop/Python
user@DESKTOP-7R05T49 MINGW64 ~
$ git config --global user.name "do-not-do-that"

user@DESKTOP-7R05T49 MINGW64 ~
$ git config --global user.email "메일주소!"

user@DESKTOP-7R05T49 MINGW64 ~
$ pwd
/c/Users/user

user@DESKTOP-7R05T49 MINGW64 ~
$ ls
'3D Objects'/
AppData/
'Application Data'@
bee-box_v1.6/
'Cisco Packet Tracer 6.2sv'/
'Cisco Packet Tracer 7.2.2'/
Contacts/
Cookies@
Desktop/
Documents/
Downloads/
eclipse-workspace/
Favorites/
Intel/
IntelGraphicsProfiles/
KakaoTalk/
Links/
'Local Settings'@
MicrosoftEdgeBackups/
Music/
'My Documents'@
NetHood@
Notepad++/
NTUSER.DAT
ntuser.dat.LOG1
ntuser.dat.LOG2
NTUSER.DAT{f04c76db-1373-11e9-85dc-b85a80fdd58f}.TM.b1f
NTUSER.DAT{f04c76db-1373-11e9-85dc-b85a80fdd58f}.TMContainer00000000000000000001.re
gtrans-ms
NTUSER.DAT{f04c76db-1373-11e9-85dc-b85a80fdd58f}.TMContainer00000000000000000002.re
gtrans-ms
ntuser.ini
OneDrive/
Pictures/
PrintHood@
PycharmProjects/
```

저는 깃허브 이름이 do-not-do-that이라 저렇게 해줬습니다.

입력을 다하셨으면 pwd 명령을 쳐서 현재 자신이 어떤 위치에 있는지 파악합니다.

저는 /c/Users/user에 있네요!

이후 ls 명령어를 이용해 현재 이 위치에 어떤 것들이 있는지 확인해봅시다.

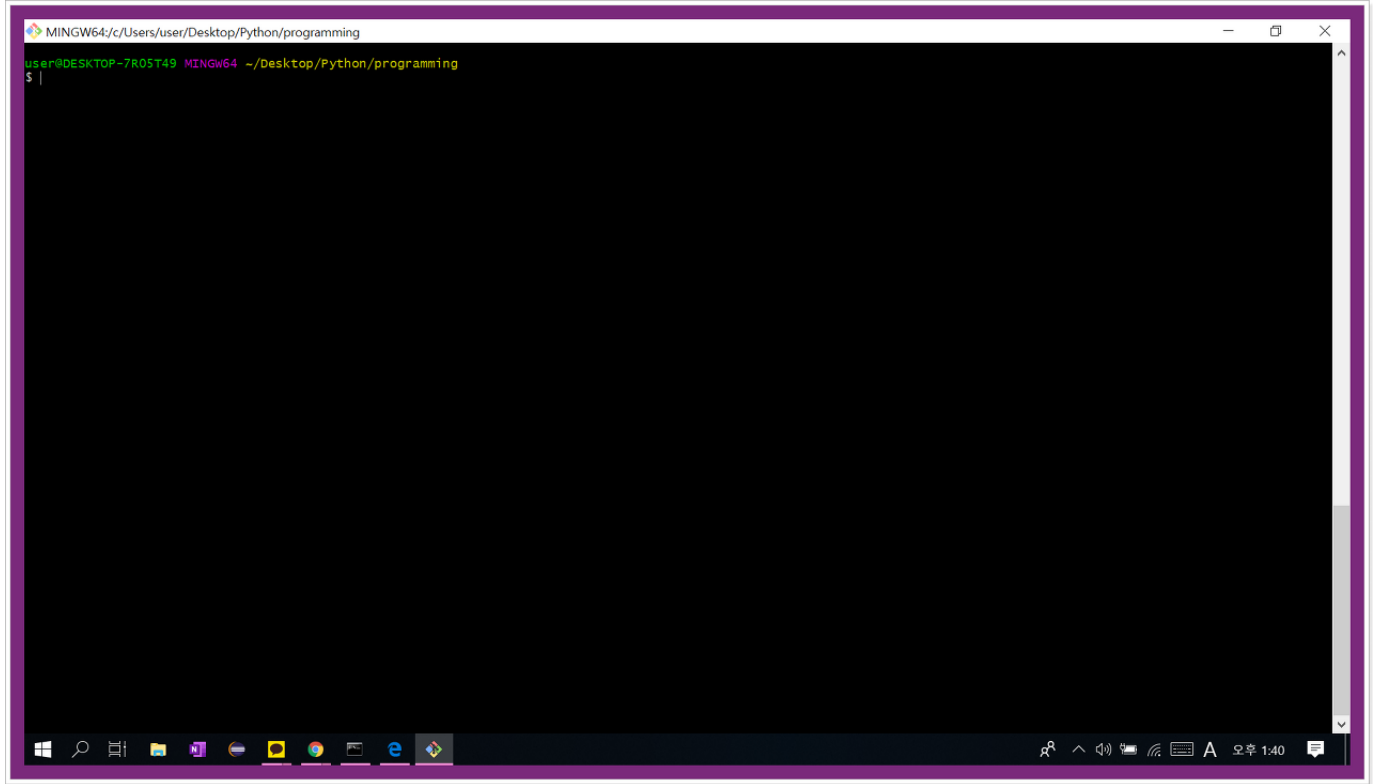
GitHub에 있는 저장소 가져오기

우선 저장소를 둘 위치로 이동을 해볼게요.

저 같은 경우 Desktop 밑에 Python 밑에 programming이라는 파일에 저장소를 둘 거예요!

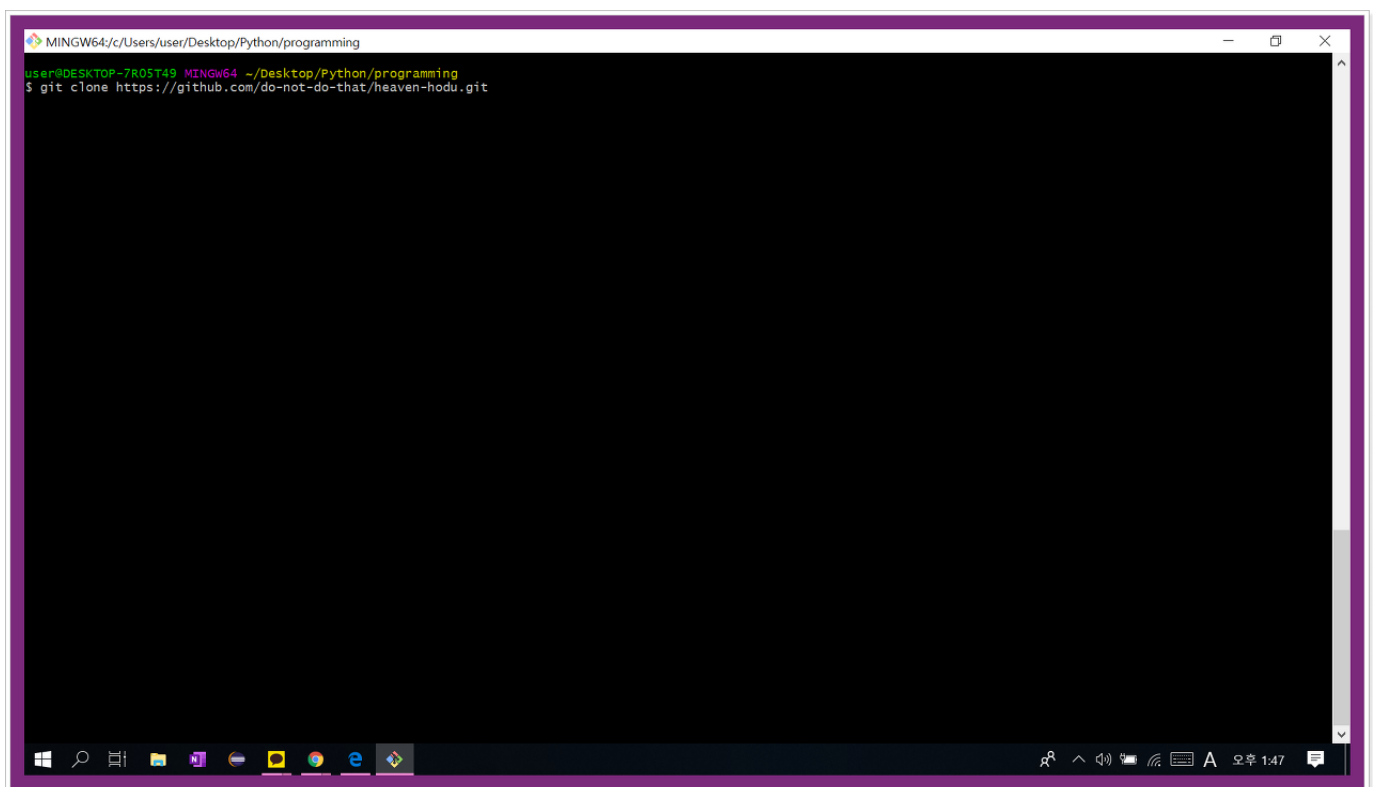
cd 명령어를 사용해서 거기까지 내려가 줍니다.

cd ./Desktop/Python/programming



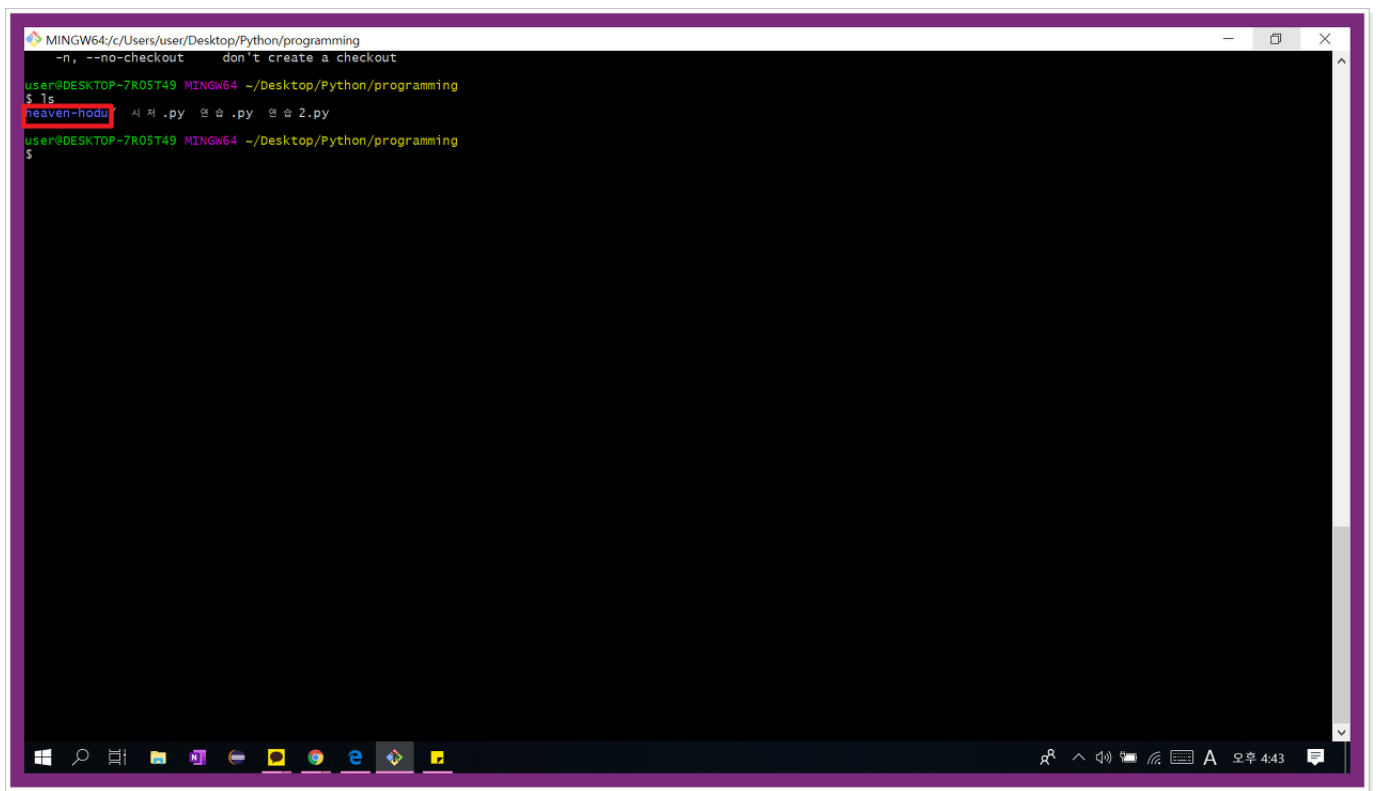
이후 `git clone` (Repo 주소)를 입력합니다.

예시를 보시죠!

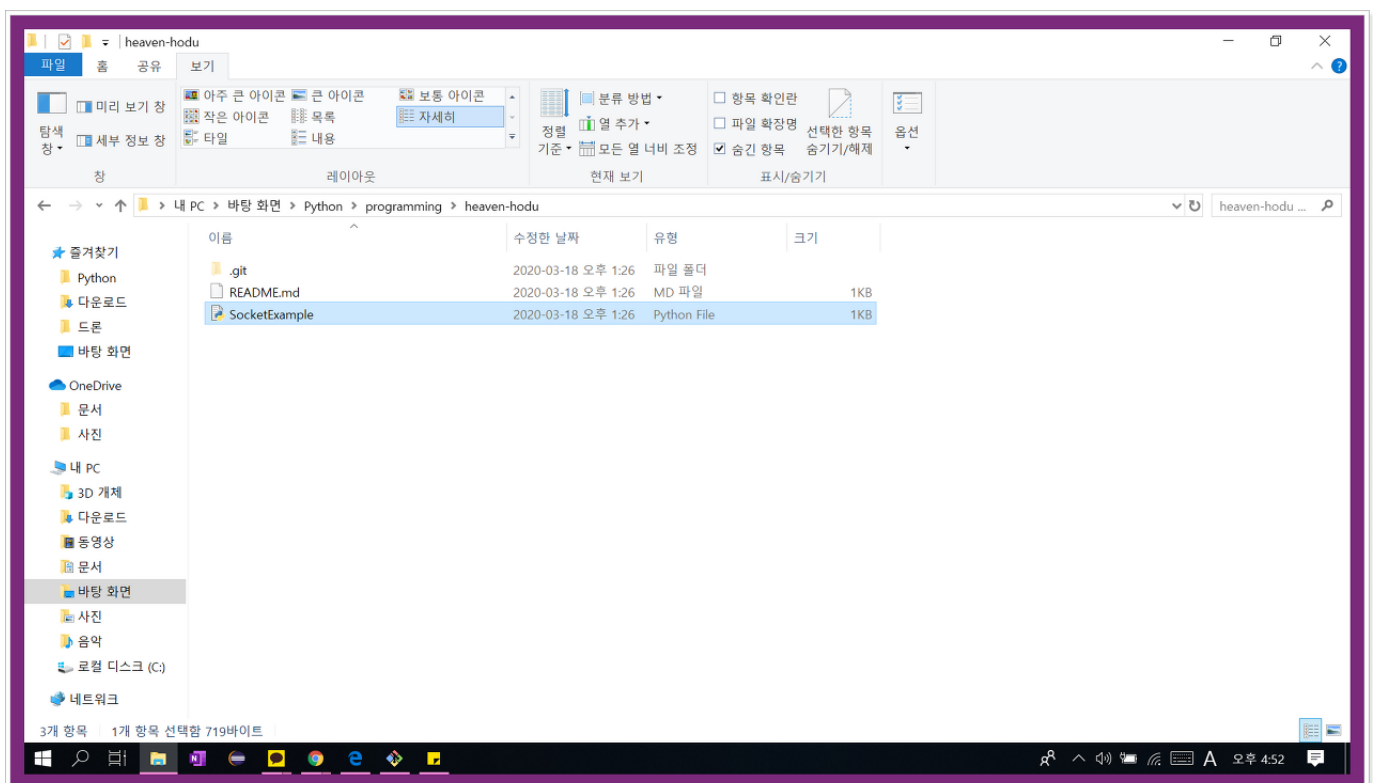


이런 식으로 저장소를 가져옵니다.

ls 명령어를 입력해서 성공적으로 저장소를 가져온 것을 확인할 수 있습니다.

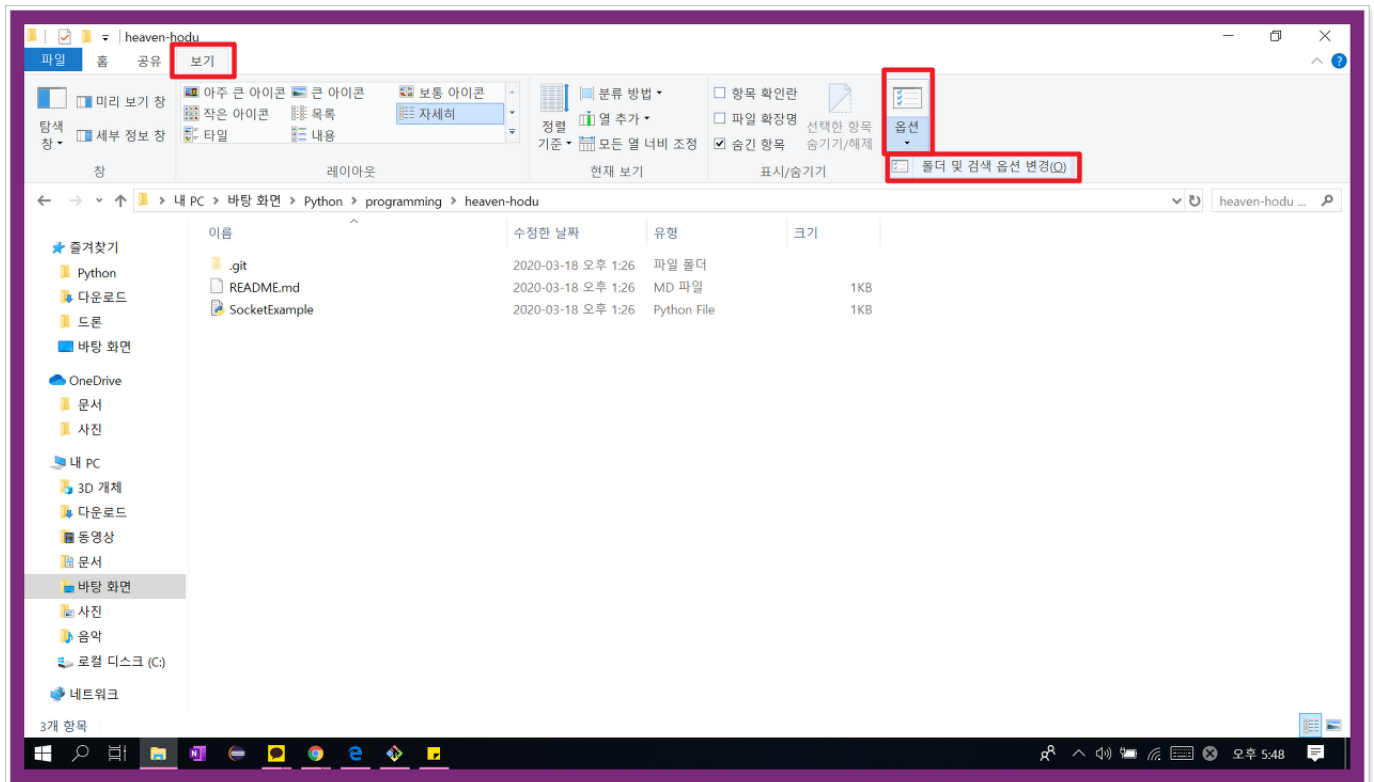


실제로 올라왔는지 들어가 볼게요.

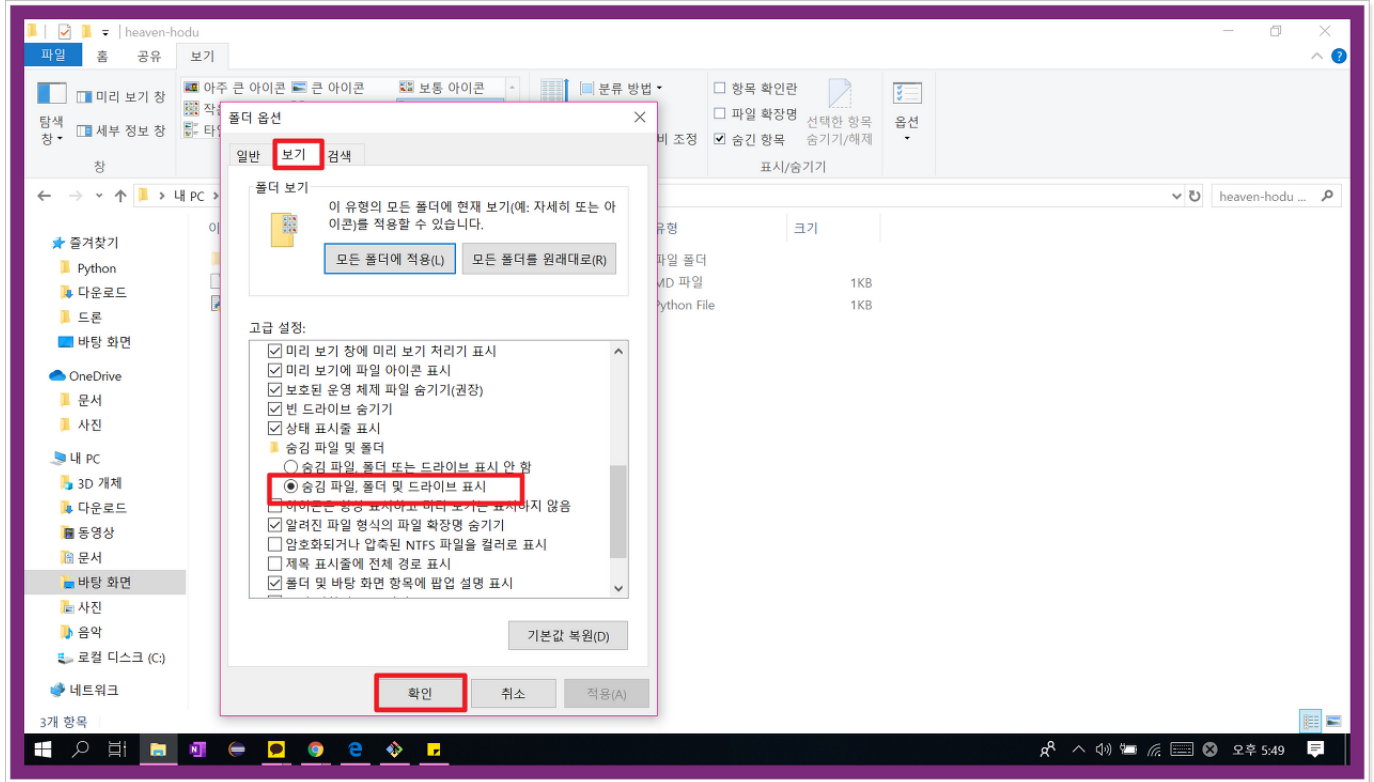


파일이 생성된 것을 확인할 수 있습니다.

안의 .git파일이 보이지 않는다면



보기 > 옵션 > 폴더 및 검색 옵션 변경으로 들어가서 폴더 옵션 창을 엽니다.



이후 보기 > 숨김 파일, 폴더 및 드라이브 표시 체크 > 확인

하면 깃 파일이 보입니다. 이걸 삭제하면 안 돼요!

Git 간단한 명령어

이 명령어들은 cmd 창에서도 되고, git bash에서도 실행이 됩니다.

원격 저장소(github 안에 있는)와 로컬 저장소 (내 컴퓨터에 있는)를 오가면서 개발을 할 때 유용합니다.

- **git init** : 현재 디렉토리를 Git 저장소로 변환해줍니다.
- **git add** : 파일을 원격 저장소에 추가합니다. (예시로 test1.py를 추가하려면 `git add test1.py` 실제 추가가 아니라 깃의 저장소의 스냅샷에 포함된다고 생각하면 될듯 합니다.)
- **git commit** : 디렉토리의 변경과 추가를 저장소에 기록합니다. (`git commit -m "New File"` : 커밋 시 남길 메시지)
- **git push** : 로컬 저장소의 변경사항을 github에 반영합니다. (`git push origin master`)
- **git checkout** : 현재 위치하고 있지 않은 저장소를 체크아웃합니다. (예를 들어 master 브랜치를 보고 싶으면, `git checkout master`를 사용할 수 있습니다.)
- **git merge** : 브랜치에서 하던 작업을 끝내고, 동료가 볼 수 있는 master브랜치로 합치는 과정입니다.
- **git pull** : 로컬 저장소 작업할 때, 작업하고 있는 저장소의 최신 버전을 받아옵니다.

실제로 로컬에서 작업한 작업물을 원격 저장소로 옮기는 방법은 이렇습니다.

cmd창을 열어 저장할 파일이 있는 파일로 이동합니다.

git init

을 이용해서 원격저장소와 연결할 수 있는 git파일을 생성하고,
로컬에 올리고 싶은 파일을

git add '파일명'

ex: git add test.py

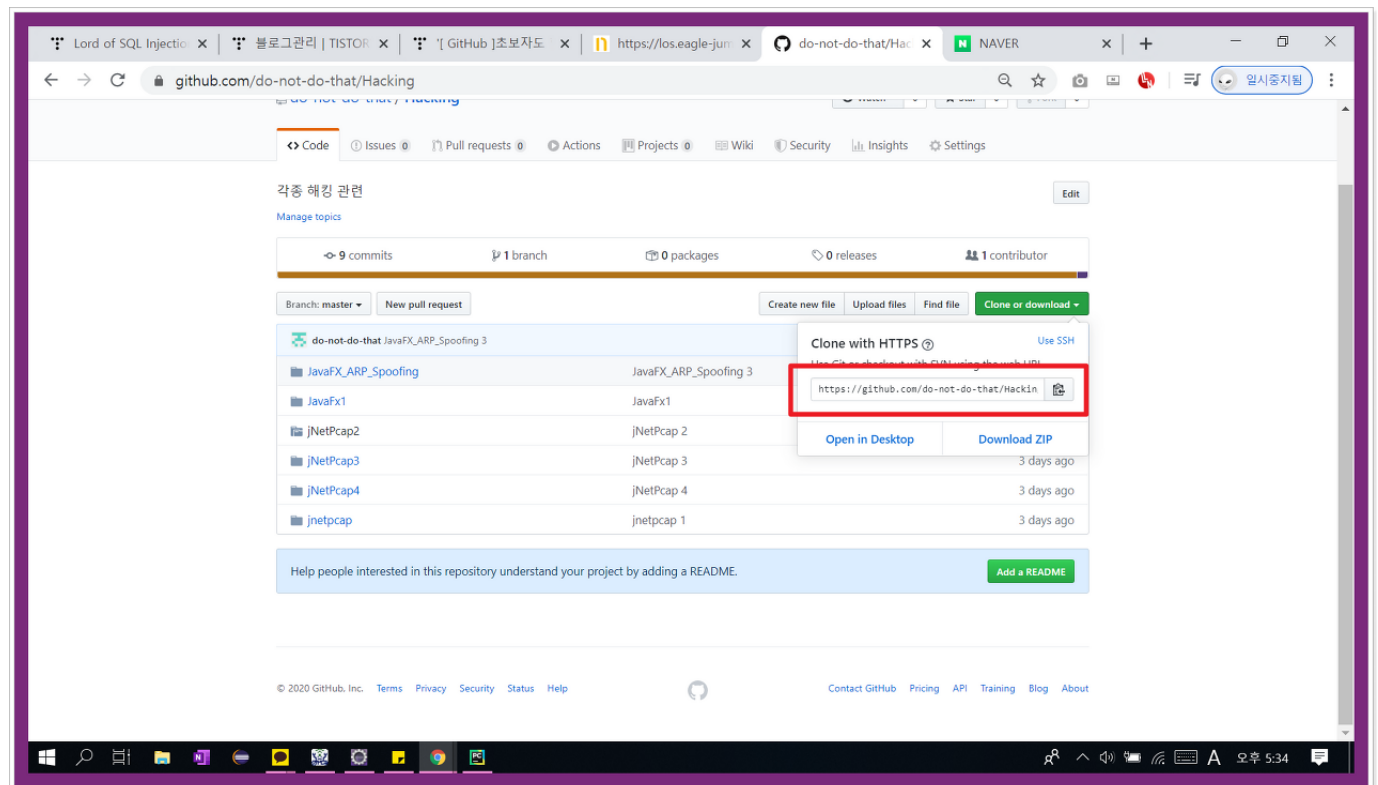
를 이용해서 git이 그 파일을 지켜보게 합니다.

git commit -m "first commit"

이후 commit을 해줍니다.

git remote -v (깃허브 remote 상태 보는 명령어)

를 입력해서 아무것도 뜨지 않으면



자신의 repo에 들어가서 위에 보이는 링크를 복사하고,

```
git remote add origin '복사한 링크'
```

를 사용해서 remote를 완료해줍니다.

ex:) git remote add origin <https://github.com/do-not-do-that/Hacking.git>

이후에

```
git push -u origin master
```

를 이용해서 로컬 저장소에 있는 파일을 원격으로 push 해줍니다.

원격에서 확인해보면, 파일이 성공적으로 올라간 것을 확인해 볼 수 있습니다.

아직 배워가는 단계이고 깃허브를 접한 지 얼마 되지 않아서
정제되어 있지 않은 지식의 포스팅입니다.

수정할 사항이나 틀린 점이 있다면 댓글로 알려주시면 감사하겠습니다.

실전 연애 활용 100%! 연애유튜버 미란다TV만의 '데이팅 가이드북'



#깃허브 사용법 #github 사용법 #깃허브 시작 #깃허브란 #github 사용 #github 원격 #깃허브 쓰는법 #깃허브 초보자

FU11M00N

방문 해주셔서 감사합니다.

구독하기 +



Git 의 다른 글

더보기

[git & github] git 설치 하기 & git 기본 사용법

이 포스팅은 구름 에듀 강좌 나동빈님의 강의 "Git으로 시작하는 협업 및 오픈소스 프로젝트" 를 기반으로 작성되었습니다. edu.goorm.io/learn/lecture/11528/git으로-시작하는-협업-및-오픈소스-프로젝트/info 구름EDU - 모두를 위한 맞춤형 IT교육 구름EDU는 모두를 위한 맞춤형 IT교육 플랫폼입니다. 개인/학교/기업 및 기관 별 최적화된 IT교육 솔루션을 경험...

♡ 0 · 💬 0 · 2021. 2. 18.



[git & github] 오픈 소스란, 깃허브 contribute, 커뮤니티

이 포스팅은 구름 에듀 강좌 나동빈 선생님 강의 "Git으로 시작하는 협업 및 오픈소스 프로젝트" 를 기반으로 작성되었습니다. edu.goorm.io/learn/lecture/11528/git으로-시작하는-협업-및-오픈소스-프로젝트/info 구름EDU - 모두를 위한 맞춤형 IT교육 구름EDU는 모두를 위한 맞춤형 IT교육 플랫폼입니다. 개인/학교/기업 및 기관 별 최적화된 IT교육 솔루션을 경험...

♡ 0 · 💬 0 · 2021. 2. 18.



[git & github] Git 등장 배경과 장점 쓰이는이유

이 포스팅은 구름 에듀 강좌 나동빈 선생님 강의 "Git으로 시작하는 협업 및 오픈소스 프로젝트" 를 기반으로 작성되었습니다. edu.goorm.io/learn/lecture/11528/git으로-시작하는-협업-및-오픈소스-프로젝트/info 구름EDU - 모두를 위한 맞춤형 IT교육 구름EDU는 모두를 위한 맞춤형 IT교육 플랫폼입니다. 개인/학교/기업 및 기관 별 최적화된 IT교육 솔루션을 경험...

♡ 0 · 💬 0 · 2021. 2. 18.



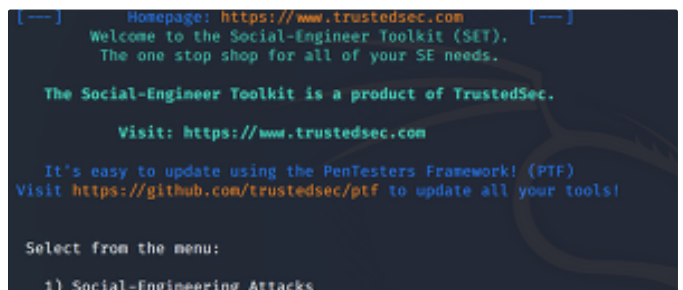
[git & github] Git 동작 원리

이 포스팅은 구름 에듀 강좌 나동빈 선생님 강의 "Git으로 시작하는 협업 및 오픈소스 프로젝트" 를 기반으로 작성되었습니다. edu.goorm.io/learn/lecture/11528/git으로-시작하는-협업-및-오픈소스-프로젝트/info 구름EDU - 모두를 위한 맞춤형 IT교육 구름EDU는 모두를 위한 맞춤형 IT교육 플랫폼입니다. 개인/학교/기업 및 기관 별 최적화된 IT교육 솔루션을 경험...

♡ 0 · 💬 0 · 2021. 2. 18.



이 블로그 인기글

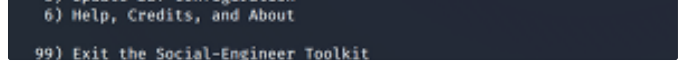


♡ 12 · 💬 6





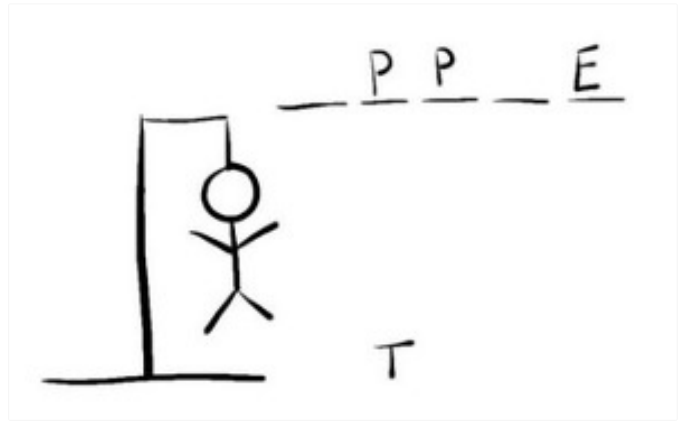
[python] socket 프로그래밍 - 다양한 데이터 보내기 (feat. pickle 직렬화)



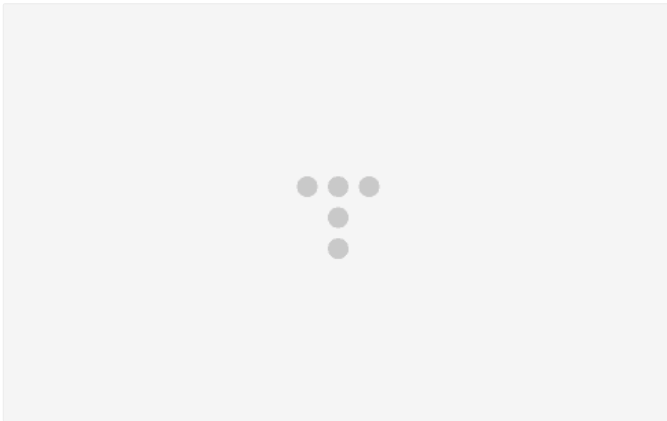
[피싱] 피싱 사이트 만들기(setoolkit)



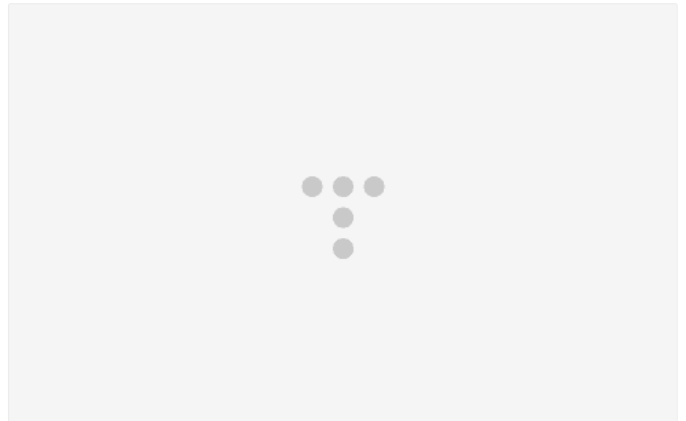
[Python] RC4 알고리즘



[C++] 임의의 문자 알아맞추기 게임 (행맨)



[React Native] ES6 Script주요 기능(화살표함수, 비구조 할당 , 리터널 함수, 객체 리터널, map 함수)



[용어] BMT, POC란?

[티스토리](#) | [로그인](#) | [고객센터](#)

티스토리는 카카오에서 ♥을 담아 만듭니다.

© Kakao Corp.