구디 아카데미  
알고리즘 스터디  
중급반

git 설정 메뉴얼 - 유준혁 작성

# **개요 및 목적**

* 새로운 PC를 사용하게 되거나, 메뉴얼을 보면서 새롭게 따라해보고 싶으신 분들을 위해.
* git의 기본적인 기능들을 혼자서 충분히 다룰 수 있도록.

# **Git 설치와 GitHub 계정 생성**

[**Git 다운로드 링크 바로가기**](https://git-scm.com/downloads)

[**GitHub 링크 바로가기**](https://github.com/)

1. 설치 링크를 눌러 Git을 다운받는다.
2. 설정값들은 모두 default로 두고 계에에에속 Next를 눌러 설치한다.
3. 설치에 시간이 좀 걸리니 그 동안 GitHub 계정을 만들어둔다.
4. 혹시나 [**백준 온라인 저지**](https://www.acmicpc.net/) 계정이 없다면 만들어둔다.

# **GitHub 개인 Repository 설정**

[**알고리즘 스터디 중급반 깃헙 Repo**](https://github.com/J511Y/GD_algostudy)

1. 공유 repo 링크를 들어가 오른쪽 위의 Fork를 누른다. ( fork를 안했을 시 )
2. 본인 Repository에 정상적으로 추가 됨을 확인하면 된다.

# **Git 개인 PC 설정**

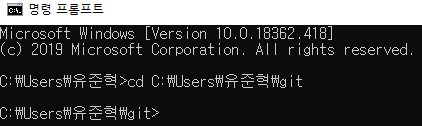
1. Git이 전부 설치가 되었다면 cmd 창을 열어 git help를 입력한다.
2. git 명령어들이 정상적으로 출력이 된다면 제대로 설치가 된 것이다.
3. git 에서는 한글 폰트를 지원하지 않기 때문에 추가적인 설정이 필요하다.

git config --global core.quotepath false

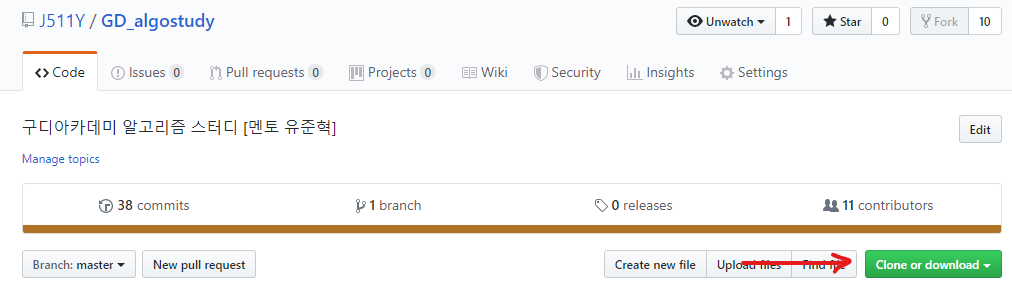
1. 앞으로 사용하기에 적당한 위치에 폴더를 생성하고 cmd도 동일한 위치에 들어간다.

ex ) c\사용자\git 이라는 폴더를 생성하고 폴더로 들어가 경로를 복사한다.

<<경로를 복사하여



cmd 창에 cd “붙여넣기” 해주면 해당 폴더로 들어가진다.

1. 개인 Git Repository에서 HTTPS Link를 복사한다.  
   
2. cmd 창에서 **git clone ‘붙여넣기’** 를 입력한다.

ex) **git clone https://github.com/본인닉네임/GD\_algostudy.git** (예시입니다)

1. 개인 Repository의 모든 파일과 폴더가 현재 위치에 생성됩니다.

(폴더 이름은 Repository 이름과 동일합니다. )

1. cmd 창에서 **cd GD\_algostudy** 를 입력해 해당 폴더로 들어갑니다.
2. 공유 Repository의 HTTPS Link를 복사합니다.

https://github.com/J511Y/GD\_algostudy.git ← 이걸 복사하셔도 됩니다.

1. **git remote add upstream https://github.com/J511Y/GD\_algostudy.git** 를 입력합니다.

git remote -v 를 입력했을 때 origin 본인계정, upstream 공유계정 이 나오면 됩니다.

1. **git config user.name [본인 깃헙 닉네임]** 을 입력합니다.

ex) git config user.name J511Y (예시입니다. 본인 계정의 닉네임을 입력하세요.)

1. **git config user.email [본인 깃헙 이메일]** 을 입력합니다.

ex) git config user.email [dbckdgur12@naver.com](mailto:dbckdgur12@naver.com)  
(예시입니다. 본인 계정의 이메일을 입력하세요.)

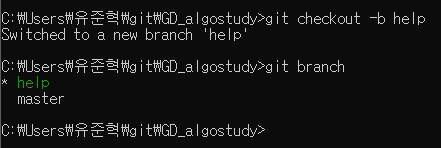
1. 기본적인 준비는 끝났습니다. 이제 제대로 동작하길 바라며 본체를 향해 큰 절을 올립니다.

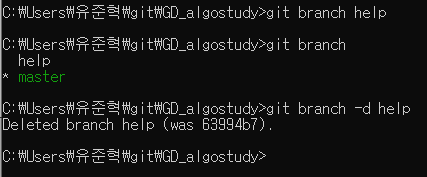
# 실제 작업 순서

*기본 세팅을 마치고 작업 파일 commit 과 pull request를 날리는 과정입니다.*

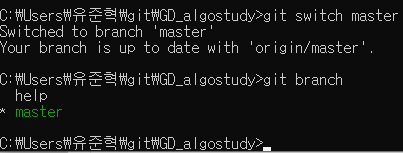
1. **git pull upstream master**를 입력해 가장 최근의 upstream을 받아옵니다.
2. master 브랜치는 공유계정과 통신할때에만 사용합니다. 새로운 브랜치를 만듭시다.
3. **git checkout -b ‘문제번호’** 를 입력합니다. ex) git checkout -b 1234

**checkout -b** 명령어는 새로운 브랜치를 만들고 해당 브랜치로 switch 해주는 명령어입니다.

  
branch를 만들고만 싶으시다면 **git branch 브랜치명** 을 입력하시면 됩니다.



branch를 이동하고 싶다면 **switch 브랜치명** 을 입력하시면 됩니다.



1. 작업한 파일을 본인 폴더에 복사해놓은 뒤 cmd 창에 **git add \*** 를 입력합니다.

원래는 git add 파일명 이런식으로 해야 되지만 스터디에선 충돌 염려가 없다고 생각이 들어 ‘모든 것’을 뜻하는 ‘\*’ 로 작업한 모든 파일을 staging 상태로 만듭니다. (새로운 파일을 commit 할 준비를 하는 과정. 아직 commit은 아님)

1. **git commit -m “남기고싶은 메세지 입력”** 을 입력합니다.

기본 양식은 git commit -m “[1주차\_조건/반복문][유준혁][백준\_1234]” 입니다.  
[ 몇주차인지 \_ 어떤 단계인지 ][ 본인 이름 ] [ 문제사이트 \_ 문제 번호 ]

1. **git push origin [브랜치명]** 을 입력합니다.

ex) git push origin 1234

1. 본인 GitHub Repository 페이지로 이동해 연노랑색으로 알림이 떠 있는 버튼을 누릅니다.
2. upstream(공유 계정) 에게 Pull Request를 작성합니다.

제목은 기본 양식을 가져가고 내용에 추가적으로 작성이 필요한 걸 적습니다.

**\*원래는 항상 commit 하기 전에 git fetch upstream master 를 입력해 upstream 과 동기화를 시켜주어야합니다.**

**하지만 저희는 충돌이 일어날 이유가 없다고 생각해 굳이 하지는 않으려고 합니다.**

**fetch와 pull의 차이는 merge까지 해주느냐 정도로 알고 계시면 됩니다. (pull이 merge까지 해주는 명령어)**

# 

# 

# 명령어 모음

***기본적인 명령어들을 모아봤습니다. 헷갈릴때면 보세요.***

git remote -v : 현재 remote 상태 보기

git remote add ‘remote이름’ ‘주소’ : remote 추가하기. 매개변수로 변수명과 주소를 받는다.

git remote remove ‘remote이름’ : remote 제거하기. origin도 제거 후 새로 설정 가능하다.

git branch : 브랜치의 목록을 보여줍니다.

git branch ‘브랜치명’ : 새로운 브랜치를 만듭니다. 기존에 존재한다면 오류가 발생합니다.

git branch -d ‘브랜치명’ : 해당 브랜치를 삭제합니다.

git switch ‘브랜치명’ : 해당 브랜치로 바꿉니다.

git add \* : 변경된 파일이 있다면 모두 staging 상태로 이동시킵니다. (commit 전 실행 명령어)

git commit -m “메세지” : staging 상태에 있는 파일들을 commit 합니다.

git push ‘remote이름’ ‘branch이름’ : branch이름에 있는 것들을 remote이름으로 push하기

git pull ‘remote이름’ ‘branch이름’ : remote이름의 해당 branch를 pull합니다.(merge까지 같이)

git config user.name ‘깃헙닉네임’ : 계정을 새로이 설정합니다.

git config user.email ‘깃헙이메일’ : 게정을 새로이 설정합니다.