

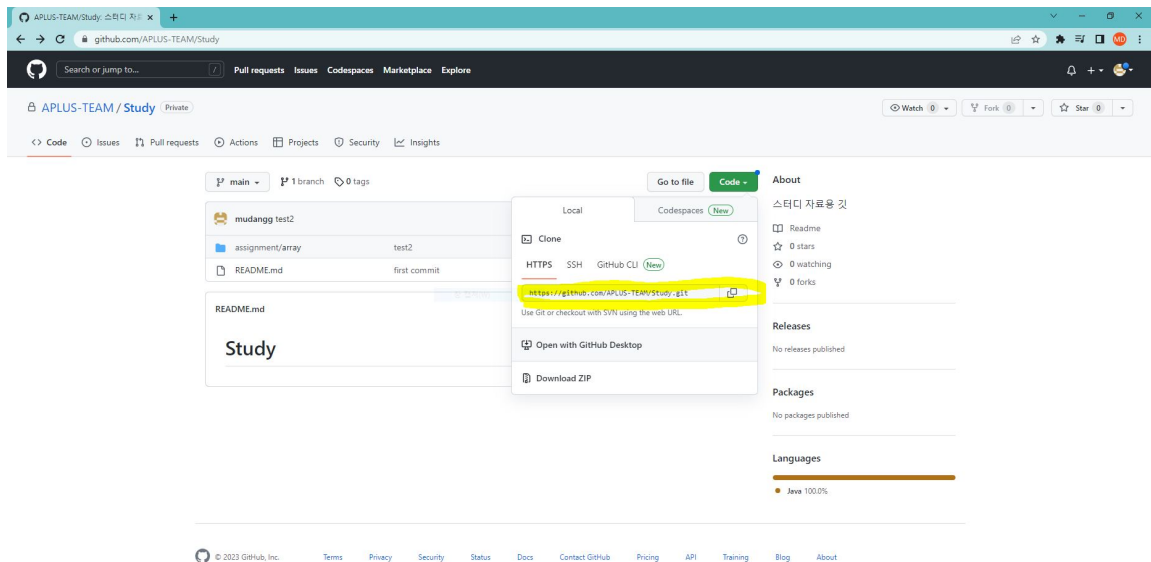
# Git 사용법

## 명령어 요약

- `git add (파일이름)` : stage에 파일 추가
    - `git add .` : 현재 디렉토리에 있는 모든 파일, 폴더를 stage에 추가
  - `git commit -m "텍스트"` : stage area에 있는 파일들을 repository에 커밋 메시지와 함께 commit
  - `git push` : 로컬 레포지토리(내 pc)에 commit된 내용을 원격 레포지토리(깃허브)에 등록
  - `git pull` : 원격 레포지토리에서 업데이트된 내용 로컬로 받아오기
  - `git clone (원격 레포 URL) a` : 원격 레포지토리 내용을 a 폴더에 생성
  - `git branch` : branch 목록 확인
  - `git switch 'A'` : 현재 branch를 A branch로 변경
  - `git switch -c 'A'` : 현재 branch를 A branch로 생성하며 변경
  - `git branch -d 'A'` : A branch 제거 (다른 브랜치에서 있을 때 명령해야함)
- 

## git clone부터 branch push까지

1. 깃허브 레포지토리로 가서 원격 레포지토리의 URL을 복사한다.



2. 복사한 URL을 이용하여 원하는 디렉토리에 git clone한다. (레포지토리 내용 다운받기)

```

MINGW64/c/Users/zune1/Desktop
$ cd Desktop/

김 승 F + 니 @DESKTOP-LJN39FC MINGW64 ~/Desktop
$ ls
'CAD LISP'/          desktop.ini
Connect/            'git 작업'/
Desktop.tar          java-project-real-final/
'GitHub Desktop.lnk'* movie/
JPEG.zip             stack.pdf
Java_PJT_v1/         study/
PS/                  zune1408_gmail.com_김 봉 준 .zip
'Photoshop CS6_ko'/  관 통 /
Practice/            '새 폴 더 '/
aaa.jpg              '새 폴 더 (2)'/

김 승 F + 니 @DESKTOP-LJN39FC MINGW64 ~/Desktop
$ git clone https://github.com/APLUS-TEAM/Study.git APLUSStudy
Cloning into 'APLUSStudy'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 11 (delta 0), reused 11 (delta 0), pack-reused 0
Receiving objects: 100% (11/11), done.

김 승 F + 니 @DESKTOP-LJN39FC MINGW64 ~/Desktop
$
  
```

APLUSStudy라는 폴더로 레포지토리를 다운받았다.

3. git branch : 현재 브랜치 존재하는 브랜치 확인 (-r 속성과 -v 속성은 추후에..)

```
MINGW64/c/Users/zune1/Desktop/APLUSStudy
'Photoshop CS6_ko'/'새 폴더'/'
Practice/'새 폴더 (2)'/
aaa.jpg

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop
$ cd APLUSStudy/

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ ls
README.md  assignment/

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch
* main

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch -r
origin/HEAD -> origin/main
origin/main

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch -v
* main 85df349 test2

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ |
```

4. git branch bongjun 명령어를 통한 bongjun 브랜치 생성 (하늘색 박스)

```
MINGW64/c/Users/zune1/Desktop/APLUSStudy

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ ls
README.md  assignment/

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch
* main

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch -r
origin/HEAD -> origin/main
origin/main

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch -v
* main 85df349 test2

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch bongjun

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git switch bongjun
Switched to branch 'bongjun'

김 승 F+ 나 @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (bongjun)
$ |
```

5. git switch bongjun 명령어를 통한 main → bongjun 브랜치 이동 (연보라 박스)

```
MINGW64:/c/Users/zune1/Desktop/APLUSStudy
김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ ls
README.md  assignment/

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch
* main

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch -r
origin/HEAD -> origin/main
origin/main

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch -v
* main 85df349 test2

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git branch bongjun

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git switch bongjun
Switched to branch 'bongjun'

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (bongjun)
$ |
```

5. 1) git switch -c bongjun2 명령어를 통해 브랜치 생성과 동시에 변경 가능

```
MINGW64:/c/Users/zune1/Desktop/APLUSStudy
$ git branch bongjun

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git switch bongjun
Switched to branch 'bongjun'

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (bongjun)
$ git switch main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git switch bongjun
Switched to branch 'bongjun'

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (bongjun)
$ git switch main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (main)
$ git switch -c bongjun2
Switched to a new branch 'bongjun2'

김 승 F ㅏ ㅓ @DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy (bongjun2)
$ |
```

6. APLUSSudy의 assignment 폴더에 Main.java 파일을 넣고 add부터 push



```
MINGW64:/c/Users/zune1/Desktop/APLUSStudy/assignment
Main.java array/
김승민@DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy/assignment (bongjun)
$ git add Main.java

김승민@DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy/assignment (bongjun)
$ git commint -m "branch_practice"
git: 'commint' is not a git command. See 'git --help'.

The most similar command is
    commit

김승민@DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy/assignment (bongjun)
$ git commit -m "branch_practice"
[bongjun 03ca712] branch_practice
1 file changed, 36 insertions(+)
create mode 100644 assignment/Main.java

김승민@DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy/assignment (bongjun)
$ git push
fatal: The current branch bongjun has no upstream branch.
To push the current branch and set the remote as upstream, use

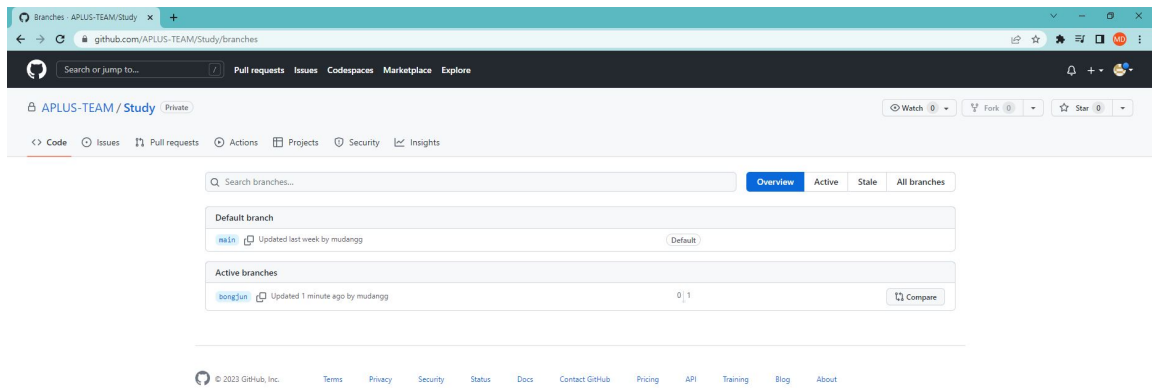
    git push --set-upstream origin bongjun

To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetupRemote' in 'git help config'.

김승민@DESKTOP-LJN39FC MINGW64 ~/Desktop/APLUSStudy/assignment (bongjun)
$
```

하지만 bongjun 브랜치에서 그냥 push해도 등록되지 않음 (원격의 bongjun branch와 어떻게 연결되는지 모르기 때문임으로 추정, 확실하진 않음)

7. git push --set-upstream origin bongjun 명령어를 입력하면, 원격 레포지토리에 bongjun 브랜치를 생성하고 연결해준다. 명령 성공 뒤 깃허브에서 확인할 수 있는 모습



**파일 이름 형식은 PS문제번호\_영어이름으로 통일합니다.**

ex) PS1234\_bongjun.java