

깃 사용법

💖 by Park yoonjeong

git version 확인

```
git --version
```

git user 확인

```
git config --list //git config list 확인

//안되었을 경우
git config --global user.name "홍길동" //ID
git config --global user.email "support@webisfree.com" //email address

//삭제하는 방법
git config --unset --global user.name
git config --unset --global user.email
```

github token 만들기

settings/developer settings/ personal access tokens

github repository 만들기

깃헙 repository에서 new해서 생성

일반적인 git 사용법 (개인 repository)

<용어>

github repository = remote repository

desktop repository = local repository

commit: remote에 업데이트하거나 받아오는 것

pull: remote에서 local로 commit을 받아옴

push: local에서 remote로 commit을 보냄

git은 버전 관리하기에 용이.

```
git init
git clone

//코드 추가된 경우
git add . //이 directory 하위의 모든 수정된 것을 add
git add <파일명/폴더명>

//변경 사항 저장 확인
git status

//commit보내기
git commit -m "message"

//commit확인
git log //나가고 싶으면 q 누르기

//remote로 commit 보내기 (github repository에 반영)
git push -u orgigin
//만약 git init으로 생성한 경우
git push <저장소명> <브랜치명>
```

```
//git add 잘못된 경우
git reset HEAD //전체 add 취소
or
git reset HEAD <파일명>
//git commit 잘못된 경우
git reset HEAD^
```

협업 시

branch 생성해줘야함.

```
//git clone해서 local repository 만들기
git clone ~~

git branch

//branch 생성
git checkout -b yj //git checkout -b <이름>

//현재 branch 확인
git branch

//git branch switch
git checkout yj //git checkout <branch이름>
```

///// 작업 중. 누군가 remote에 업데이트 해서 받아올 경우

//하던 작업을 add + commit 하기 (본인 branch에)

```
git add .  
git commit -m ""
```

//git branch main으로 switch

```
git checkout main
```

// remote 변경 사항 main branch에 받아오기

```
git pull origin main
```

// 자신의 작업본에 반영하기

```
git checkout yj  
git merge main
```

**** 항상 push 하기 전, 자신의 작업본을 본인 branch에 add, commit한 후, main branch로 바꿔 거기서 pull하고 본인 작업본에 반영하기(본인 branch에서 main merge)!!!**
그런 다음 다시 main branch로 바꿔 본인 branch merge하고 push 하기!!!

**** 같은 곳을 여러명이 동시에 수정했다면, 충돌이 생김. -> 그렇기에 항상 작업하기 전에 pull하고 작업 중간중간 add, commit후 pull하는 것을 해줘야!**

**** 웬만하면, 역할분담시 다른 파트 말도록**

**** 잘못 push해서 다 날라간 경우, 이전 버전으로 되돌리기**

```
//revert 명령어 사용  
git revert commit번호
```

Github에서 버튼을 클릭하시면 commit 이력을 보여주는 아래의 사진을 볼 수 있습니다.

Commits on Dec 02, 2017		
	wooseung	brchC에서 기능 구현중 c3fa446
	wooseung	C에서 1번째 기능 구현 8b267c2
	wooseung	C에서 2번째 기능 구현 f37a6b7
	wooseung	C에서 3번째 기능 구현 1d9184f
	wooseung	C에서 4번째 기능 구현 9728212
	wooseung	C에서 5번째 기능 구현 6d5ed94

현재 마지막 커밋 내용은 "C에서 5번째 기능 구현" 버전 (6d5ed94)입니다.

버그가 발생했기 때문에 안전하다 생각되는 곳은 "C에서 2번째 기능 구현" 버전이라 생각되어, f37a6b7 버전으로 돌아가려 합니다.

그러면 "C에서 3번째 기능 구현" 버전의 commit 번호(1d9184f)로 **revert**를 하면 됩니다.

하다가 에러나거나 모르겠으면

에러시: 그거 복붙해서 구글링

모르는 것: 키워드 구글링