|  |  |
| --- | --- |
| **명령어** | **설명** |
| git config --global user.name [user name] | 작업자 이름 설정 |
| git config --global user.email [user email] | 작업자 이메일 설정 |
| git config --global --list | 설정값(이름 및 메일등 ) 확인 |
| git init | git 저장소(repo) 만들기 |
| git remote add [remote name] [remote addres] | 별명으로 원격지주소를 저장 |
| git remote rm [remote name] | 별명의 원격지를 삭제 |
| git remote rename [remote name] [new name] | 별명을 새로운 별명으로 변경 |
| git fetch [remote name] | remote의 모든 정보를 가져옴(모든 branch) |
| git pull | 저장소에서 변경 내용 가져오기 |
| git push | commit들을 master 저장소에 저장 |
| git push [remote name] [localbranch name] | local branch의 내용을 업데이트 |
| git push [server] tag [TAG] | server에 tag 전송 |
| git push [server] --tags | 변경된 모든 tag 전송 |
| git push [server] [L.B]:[R:B] | server 에 local branch를 -Remote branch이름으로저장 |
| git tag [TAG NAME] | 저장소에 태그를 붙인다. git tag 태그목록을 본다. |
| git branch [branch name] | 저장소의 branch name으로 branch를 만든다. |
| git branch | branch 목록을 본다. |
| git branch -a | 현재 생성된 모든 local branch와 reomte branch 확인 |
| git checkout [branch name] | 다른 브랜치로 전환 |
| git checkout -b [branch name] | branch 생성 |
| git checkout [file or folder] | git repo 기준 마지막 commit 상태로 돌림 |
| git checkout [id] [file or folder] | git repo 기준 id에 해당하는 commit 상태로 돌림 |
| git checkout -f | 아직 commit 되지 않은 working tree와 -index 수정사항 모두 사라짐 |
| git merge [branch name] | branch의 내용을 가져와 합침 |
| git add [file or folder] | git에 file 또는 folder 추가 |
| git add \* | git에 모든 file 또는 folder 추가 |
| git rm [file or folder] | git 파일 또는 폴더 제거 |
| git status | 현재 git 상태 보기 |
| git commit -m [message] | message를 repo에 저장 |
| git diff | local과 remote의 차이점을 보여줌 |
| git remote | remote서버 확인 |

|  |  |
| --- | --- |
| **명령어** | **설명** |
| git add file명 | 해당 file만 index 영역에 등록한다. Untracked file을 추가할 때도 사용 |
| git add . | 수정한 모든 file들을 index 영역에 등록한다. (Untracked file 들도 모두 commit 되므로 주의 필요) |
| git rm file명 | 해당 file만 workspace에서 삭제하고 index 영역에 등록한다. |
| git add -u | 임의로 먼저 파일들을 삭제한 다음 git add -u 를 하면 Index영역에 등록 된다. (Workspace에 modified된 파일들도 Index영역에 등록되므로 주의가 필요함) |
| git commit -m “코멘트” | 현재 Index 영역에 있는 변경 내역을 commit 영역에 등록한다. |
| git commit -a -m “코멘트” | Index에 등록 안된 변경 내역들도 모두 commit 영역에 등록한다. |
| git push | Local commit을 Remote repository에 반영 시킨다. |
| git pull | Remote repository에 반영된 내역을 local repository 및 내 소스에 받아 온다. |
| git status | 현재 Add / Commit 또는 새로 추가된 파일들의 목록을 보여줌. |
| repo status | 다수의 git 프로젝트로 구성된 경우 각 git 프로젝트를 검색하여 변경파일을 출력함 |
| git checkout -f | 변경된 file들을 Head 상태로 원복 |
| git reset --hard  commitID | 지정한 commit ID 이후의 commit들을 제거함. (commit ID는 git log 를 통해 확인) |
| git checkout -- file명 | 지정한 file을 수정하기 이전 상태로 원복 |
| git reset HEAD file명 | 지정한 file을 Index stage에서 제거한다. |
| git clean -f | 현재 자신의 Work space에 있는 Untracked file들을 모두 삭제 함 |
| git branch | 현재 내가 위치한 Branch를 확인 할 수 있다. |
| git log | 현재까지 Commit된 목록 들을 확인 한다. |
| git tag | 내가 위치한 repository에 생성되어 있는 Tag 목록을 출력한다. |
| repo forall -c git 명령어 | 전체 git 프로젝트를 찾아가서 git 명령어를 수행해 준다.변경된 file들을 Head 상태로 원복 |

출처: <https://12bme.tistory.com/47?category=718952> [길은 가면, 뒤에 있다.]