\$ echo "1st version" > file1 # 새 파일 or 수정 \$ git config --system core.autocrlf true \$ echo "Add a line" >> file1 # 파일에 추가 \$ git config --system core.safecrlf false \$ cat > file2 # 여러 줄 입력(새 파일 or 수정) 사용자 설정: [사용자계정]/.gitconfig v 📙 new-repo 1st version \$ git config --global user.name hskang Add a line [Enter] hooks \$ git config --global user.email hskang@gmail.com [ctrl + C] info \$ git config --global init.defaultBranch main \$ touch file1 file2 # 빈 파일 생성 objects \$ cat file1 file2 # 파일 내용 보기 저장소 생성 ^{\$ git init} info pack 브랜치 이름 수정 \$ git init . # 현재 폴더 refs \$ git branch -m | -M new-branch heads \$ git init new-repo - tags Initialized empty Git repository in C:/git-master/new-repo/.git/ WD(Working Directory) \$ rm -rf .git # 저장소 기능 상실 \$ git init # 현재 폴더 저장소 생성 \$ rm file git add. git diff git add --all | -A git restore file git add file git rm file 저장소 설정: [basic]/.git/config \$ git config --local core.editor 'code –wait SA(Staging Area, Index) git restore --staged --worktree file git restore -S -W file \$ git rm --cached file git diff HEAD git commit -m "message" git restore --worktree | -W file git commit --am "message" git diff --staged | --cached git restore --staged | -S file \$ git log GR(Git Local Repository) \$ git log --oneline \$ git log --graph –all \$ git commit --amend \$ git log --stat -n \$ git log --patch | -p \$ git checkout main diff HEAD~2 HEAD git checkout git restore --source=HEAD~ -W file #WD에 복구 \$ git switch main git restore -s HEAD~ -W file \$ git switch -\$ git restore --source=HEAD~ -S file #SA에 복구 \$ git show HEAD~ \$ git restore -s HEAD~ -S file \$ git restore --source=HEAD~ -W -S file #WD, SA에 복구 \$ git show --oneline HEAD~ \$ git restore -s HEAD~ -W -S file \$ git show -s HEAD~ git checkout HEAD~2 브랜치 생성과 삭제, 목록 보기 git switch -<mark>d | --d</mark>etach H<mark>E</mark>AD~2 \$ git branch new-branch 전 태그 \$ git checkout -b new-branch \$ git tag v2.0.0 \$ git switch -c new-branch \$ git tag v1.0.0 HEAD~ \$ git branch --delete | -d branch-name \$ git tag -a v1.1.0 -m "1st version" 깃 저장소 \$ git branch -D branch-name \$ git tag -a v1.1.0 # 편집기에서 메시지 추가 \$-git-branch----\$ git tag -a v1.1.0 HEAD~ # 편집기에서 메시지 추가 \$ git branch -v \$ git tag -a v1.1.0 HEAD~ -m "1st version" RR(Remote Repository, Github) 저장소 복제 \$ git clone URL \$ git clone URL NEW-DIR \$ git clone URL.

간편 파일 편집과 내용 보기

전체 설정: [설치폴더]/Git/etc/gitconfig