***Configuration:***

***// user name***

git config --global user.name "username"

***//change name***

git config --global -- replace-all “user.name "username"

***// user email***

git config --global user.email "useremail"

***// transfer to the folder***

cd /folderaddress (without space in name folder)

***// set folder as repository***

git init

***//connect local repository with remote by link. URL from github***

git remote add origin {URL}

***//shows the remote repository address***

git remote -v

***//config list with all data***

git config --list

***Work with user`s repository:***

***//shows the status of all unsaved (uncommitted) changes to repository files***

git status

***//shows all commits***

git log

***//saving state, version***

**///for all files**

git add .

**///for specific files**

git add "/folder\_address"

***//Commit - remember the version and create a new unique code (hash)***

git commit -m "description or comment"

***//upload local repository data to the remote repository, where (origin=remote repository url), (master=branch)***

git push origin master

***//Changing the state or version of data, where "hash" its a unique code***

git checkout hash

***Branches***

***//the list of created branches***

git branch

***//create branch, where newbranchname=it`s new branch name***

git branch newbranchname

***//branch switching***

git checkout branchname

***//Merge (слияние одной ветки с другой - выбираем главную master и в строке указываем из какой ветки соединить данные в главную) (merging one branch with another - select the main master and indicate in the line from which branch to transfer the data to the main one)***

git merge nameofbranch

***//перемещает коммиты из данной ветки в ту на которой сейчас находится пользователь, moves commits from a given branch to the one the user is currently on***

git rebase branchname

***//Delete branch***

git push -d origin nameofbranch

***Work with another remote repository:***

***//create new local folder for connect and copy data from remote repository***

cd /folderaddress

git clone {URL}

***//download all commits which don't exist yet (last version and all history), where (origin=remote repository url), (master=branch)***

git pull origin master