**Git Basic Commands and Simple Guide**

**Install neccessary environment for Git**

* Firstly download suitable version of git from [**https://git-scm.com/**](https://git-scm.com/)
* Then you can work with **cmd** for windows or **terminal** for linux distrubitions, also git package includes **Git Bash** and **Git Gui**

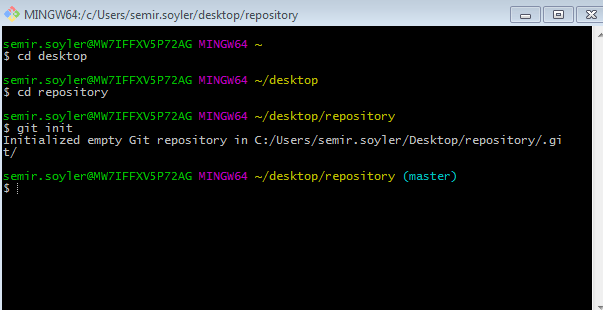
**Start to work with Git**

* **Git init** 🡪 this command provide to create new git repository

Open Git bash

Create a file for repository and go to this file in Git Bash

Then run the git init command



* **Git Clone 🡪** this command provide to get repository from remote to local or get local repository from your local as a copy

**git clone /path/repository 🡪** local copy, You can move your local repository to another directory

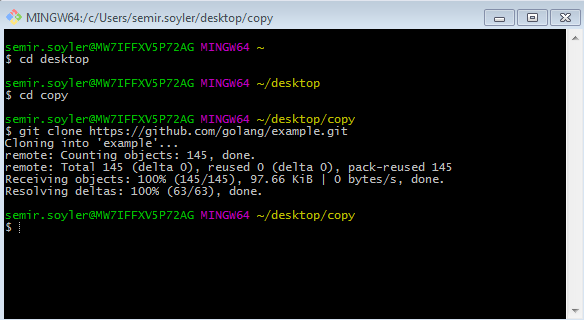
go to new directory with **cd**  command then run a command like below

**example** : git clone file:///desktop/repository/git\_file

git clone repository\_path(url of repository) or

git clone username@host:/path/to/repository (remote server )

go to directory wiht cd command then get remote repoistory

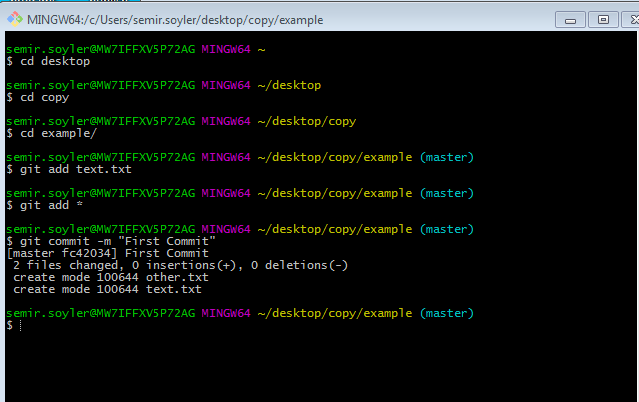


**Now you have a git repository and you can start to work**

* **git add \*** 🡪 this command adds all files to repository
* **git add specific\_file** 🡪 thia command adds just selected files
* **git commit –m “Commit Message ” 🡪** this command provide to make o commit after changing something at the project

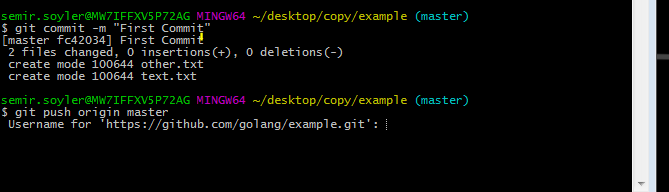
the example is below

1. go to repository
2. use git add to add files to repository (first one specific second one add all files )
3. make a commit for this changes



**Now we will send to remote server this changes**

* **git remote add origin <server> 🡪** if there is not remote server you can add with this command
* **git push origin master** 🡪 this command send changes to remote server

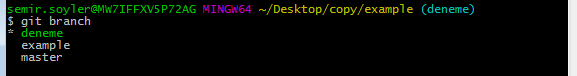


Enter the credentials (username,password, vs..) and push finish

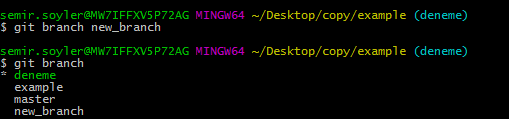
This credentials change according to your server used for repository, at this example server is github

**Working with branches**

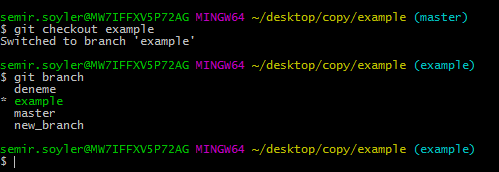
* **git branch** 🡪 this command show all branches



* **git branch new\_branch** 🡪 this command create a new branch with new\_branch name

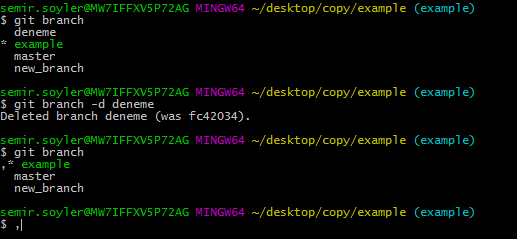


* **git checkout <another\_branch>** 🡪 this command provide to change your branch, example; **git checkout master**

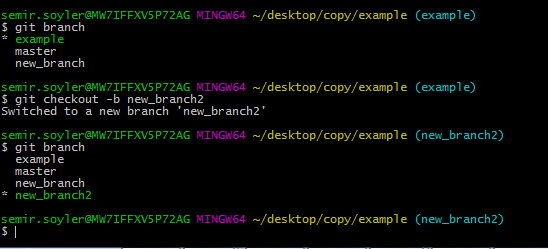


firstly I changed my branch to master then I changed branch to example

* **git push origin branch\_name** 🡪 we must send to remote new branches to use someone else also this branches
* **git branch –d branch\_name**  🡪 this command delete a branch



* **git checkout –b branch\_name** 🡪 this command create a new branch and make active this branch

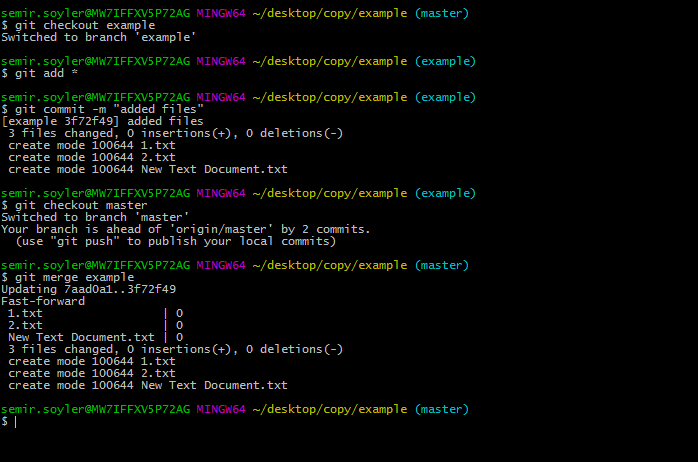


**Update and Merge**

* **git pull** 🡪 this command provide to get changes from remote to your local , \* Must be carefull about current branch to get correct changes you need

If there is not changes at remote git say 🡪 Already up-to-date.

* **git merge branchname 🡪**  this command merge branches from not active branch to active branch



1 - switched to example branch

2 - added 3 files to example branch and committed them

3 - switched to master branch

4 - merged from example branch to master branch

At above example we worked at local to send remote server these changes we can make 🡪 we can use git push origin master

* **git diff first\_branch second\_branch 🡪**  you can use this command before merge branches to see what is the difference between them
* **git checkout - - file\_name 🡪**  this command provide to retrieve changes if something done wrongly
* **git fetch origin 🡪**  this command retrieve all changes
* **git reset –hard origin/master 🡪**  and after fetch this command pull files from remote

**Summary of Commands and Some Helpful Commands**

* **gitk 🡪** run the git gui
* **git config color.ui true 🡪** colored git outputs
* **git log 🡪** shows all commits on current branch
* **git status 🡪** shows current branch , if we changes to commit or not and difference with current branch and origin/master branch