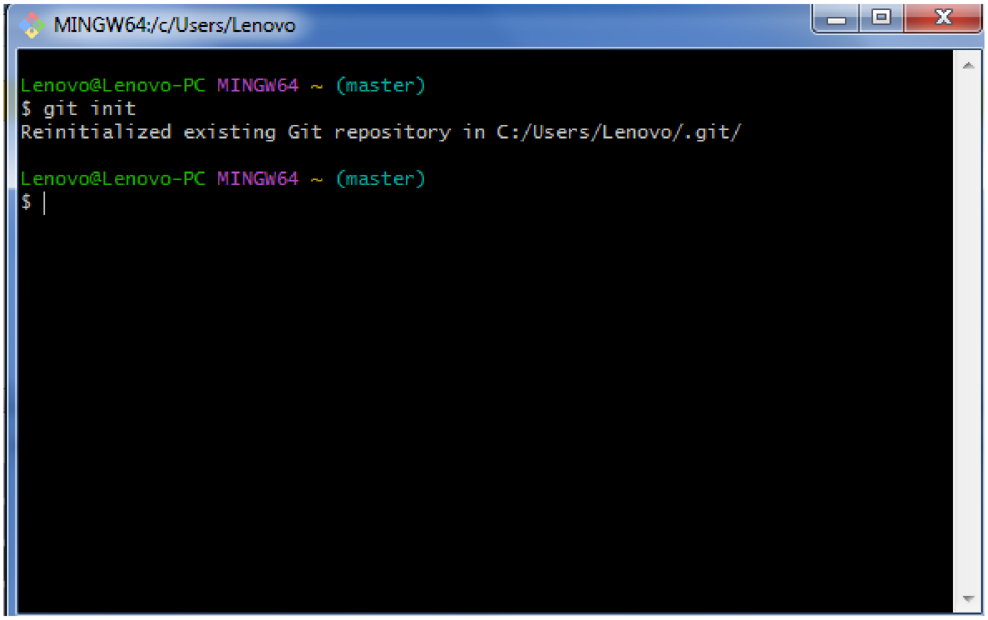
GIT COMMANDS

To perform Git operations, you typically use the Git command-line interface or a Git GUI client. Here are some common Git operations using the command-line interface:

1. Initialize a Git repository:

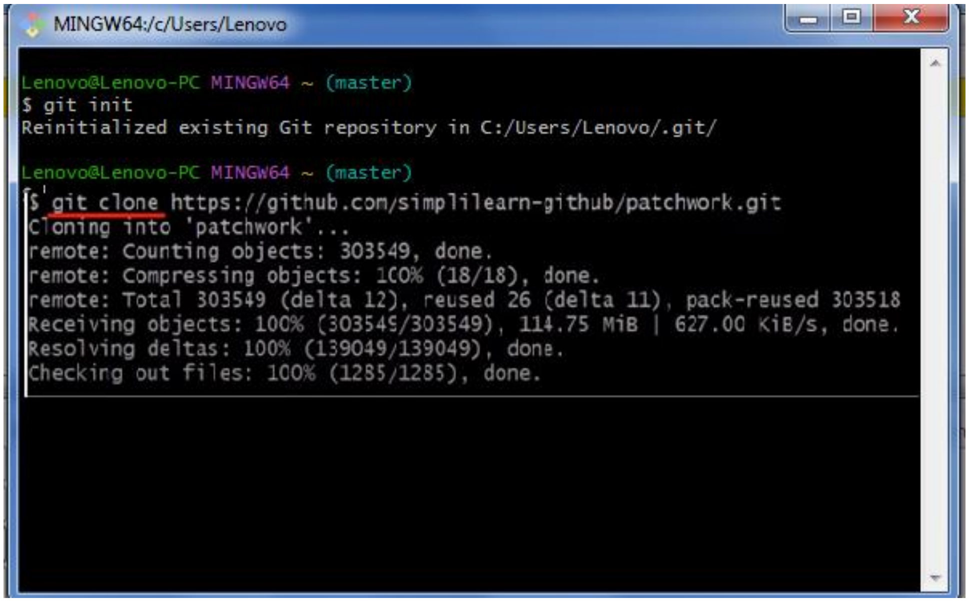
git init



```
Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/Lenovo/.git/

Lenovo@Lenovo-PC MINGW64 ~ (master)
$ |
```

2. Clone a repository:

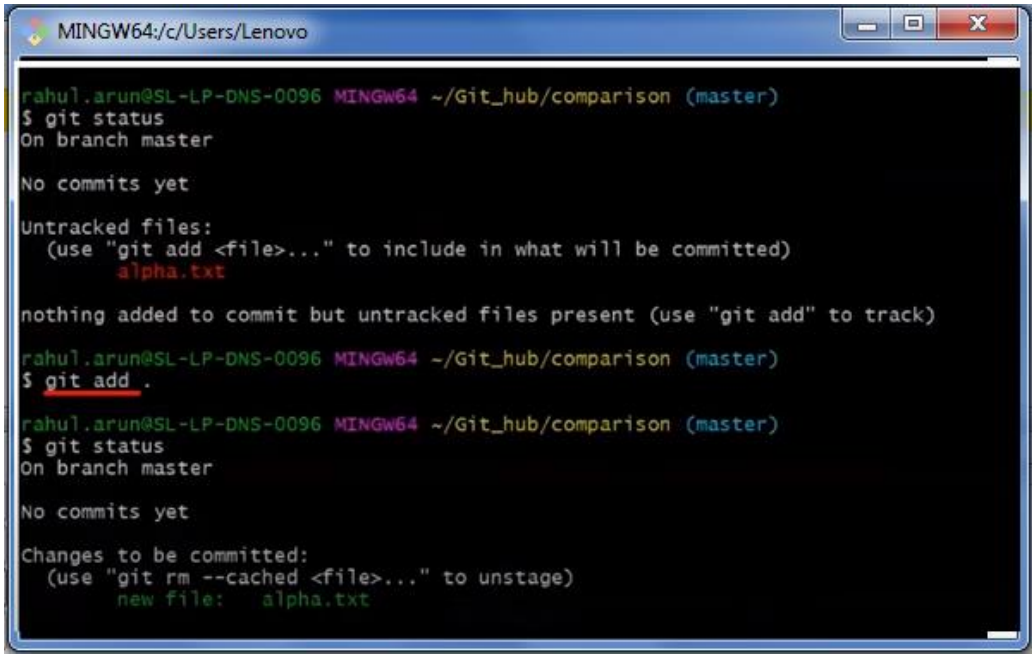
git clone https://github.com/AustinCruz04/Devops_exp-1.git



```
Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/Lenovo/.git/

Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git clone https://github.com/simplilearn-github/patchwork.git
Cloning into 'patchwork'...
remote: Counting objects: 303549, done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 303549 (delta 12), reused 26 (delta 11), pack-reused 303518
Receiving objects: 100% (303549/303549), 114.75 MiB | 627.00 KiB/s, done.
Resolving deltas: 100% (139049/139049), done.
Checking out files: 100% (1285/1285), done.
```

3. Add files to the staging area:

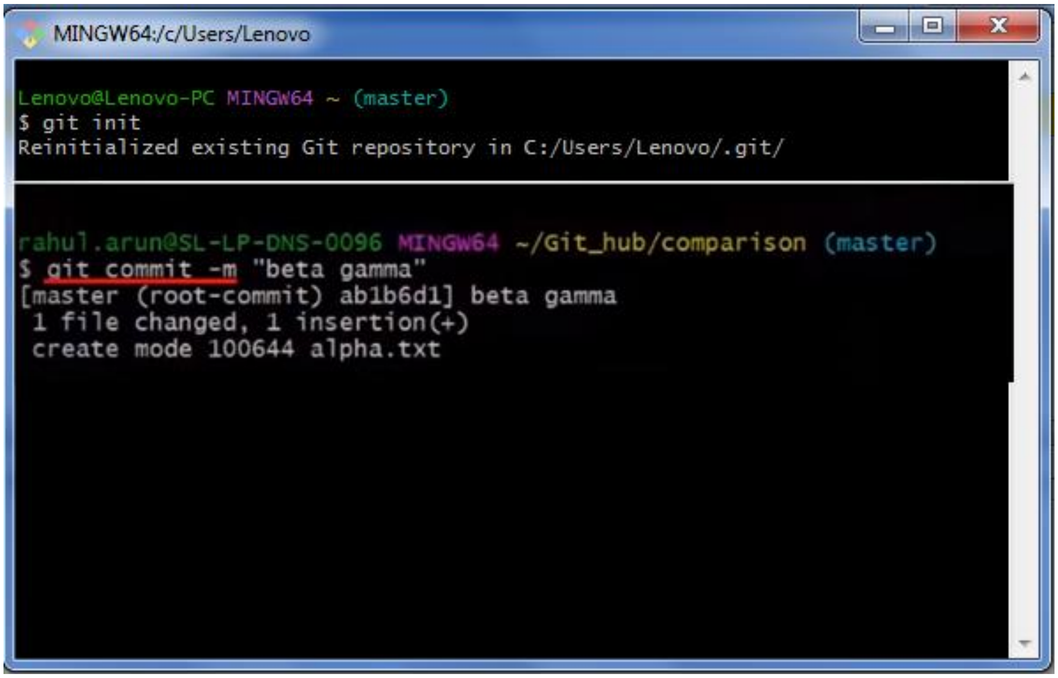
git add <file_name>



```
rahul.arun@SL-LP-DNS-0096 MINGW64 ~/Git_hub/comparison (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        alpha.txt

nothing added to commit but untracked files present (use "git add" to track)

rahul.arun@SL-LP-DNS-0096 MINGW64 ~/Git_hub/comparison (master)
$ git add .

rahul.arun@SL-LP-DNS-0096 MINGW64 ~/Git_hub/comparison (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   alpha.txt
```

4. Commit changes to the repository:

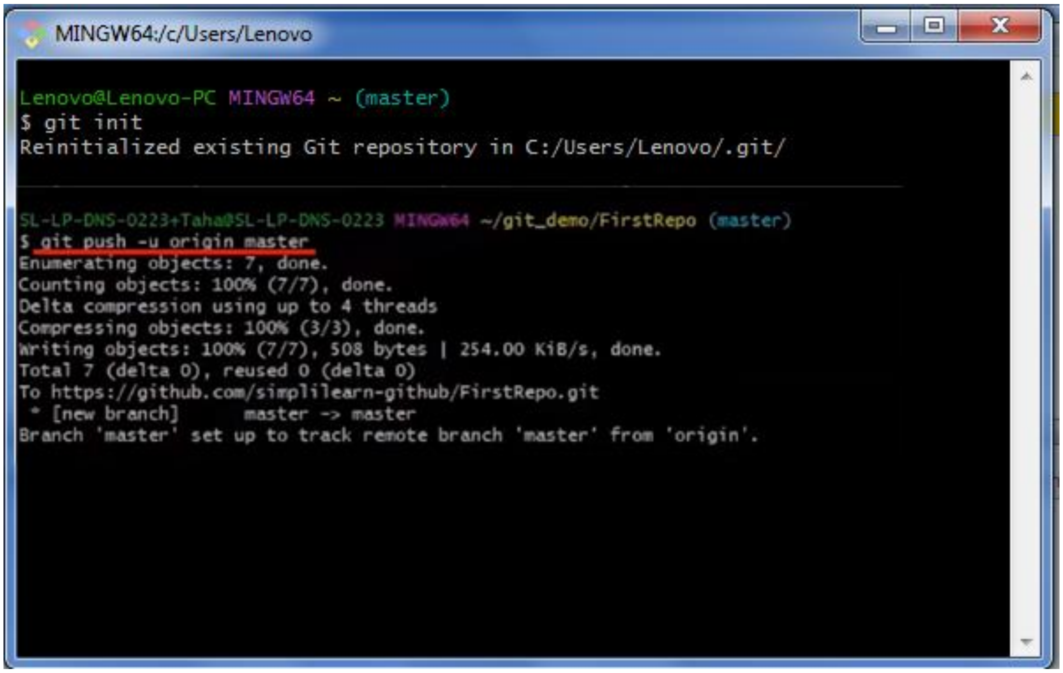
git commit -m "Commit message"



```
Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/Lenovo/.git/

rahul.arun@SL-LP-DNS-0096 MINGW64 ~/Git_hub/comparison (master)
$ git commit -m "beta gamma"
[master (root-commit) ab1b6d1] beta gamma
 1 file changed, 1 insertion(+)
 create mode 100644 alpha.txt
```

5. Push changes to a remote repository:

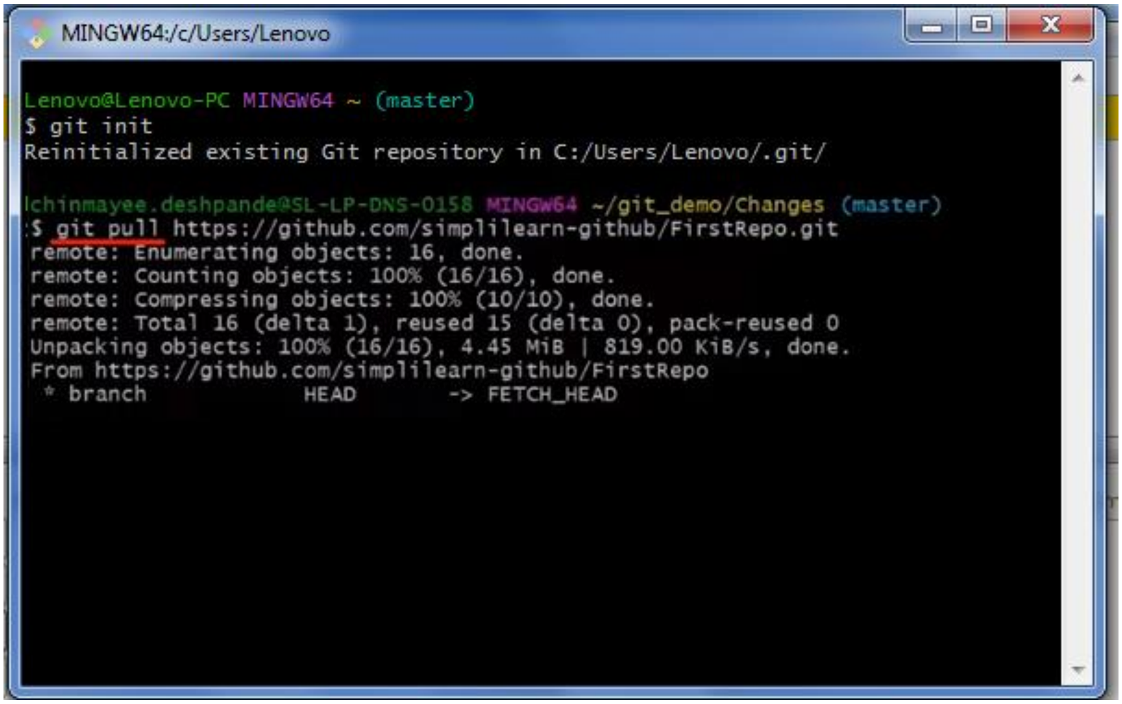
git push origin <branch_name>



```
Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/Lenovo/.git/

SL-LP-DNS-0223+Taha@SL-LP-DNS-0223 MINGW64 ~/git_demo/FirstRepo (master)
$ git push -u origin master
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (7/7), 508 bytes | 254.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0)
To https://github.com/simplilearn-github/FirstRepo.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

6. Pull changes from a remote repository:

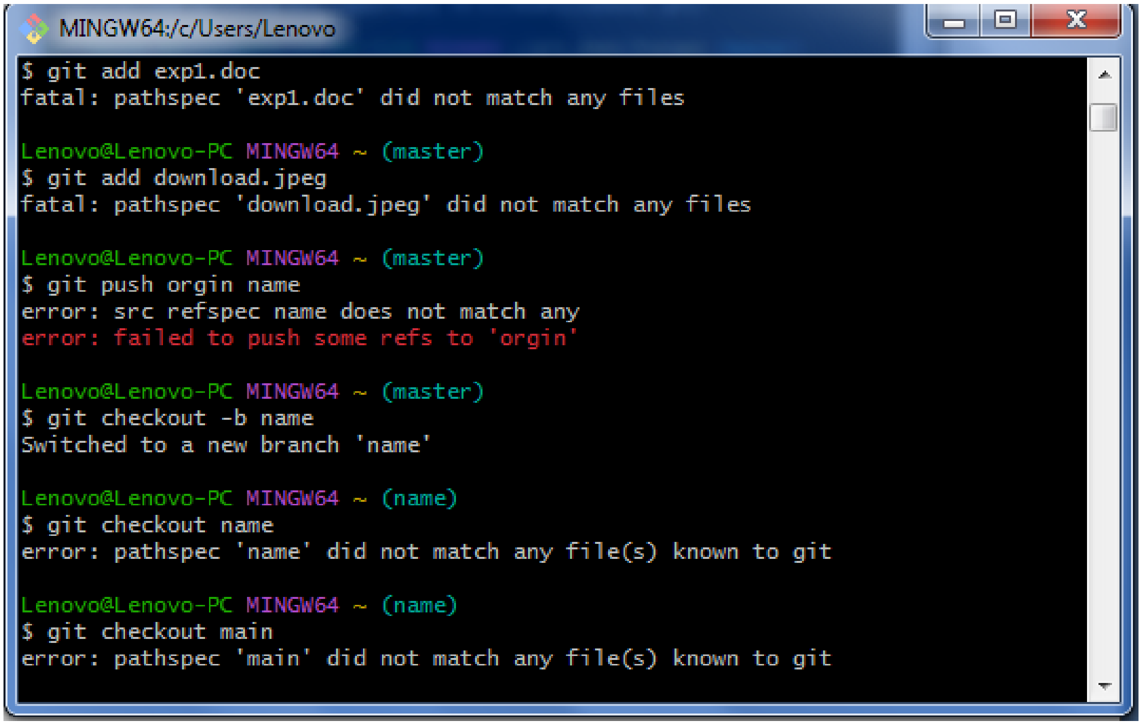
git pull origin <branch_name>



```
Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/Lenovo/.git/

lchinmayee.deshpande@SL-LP-DNS-0158 MINGW64 ~/git_demo/Changes (master)
$ git pull https://github.com/simplilearn-github/FirstRepo.git
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 16 (delta 1), reused 15 (delta 0), pack-reused 0
Unpacking objects: 100% (16/16), 4.45 MiB | 819.00 KiB/s, done.
From https://github.com/simplilearn-github/FirstRepo
 * branch            HEAD       -> FETCH_HEAD
```

7. Create a new branch:

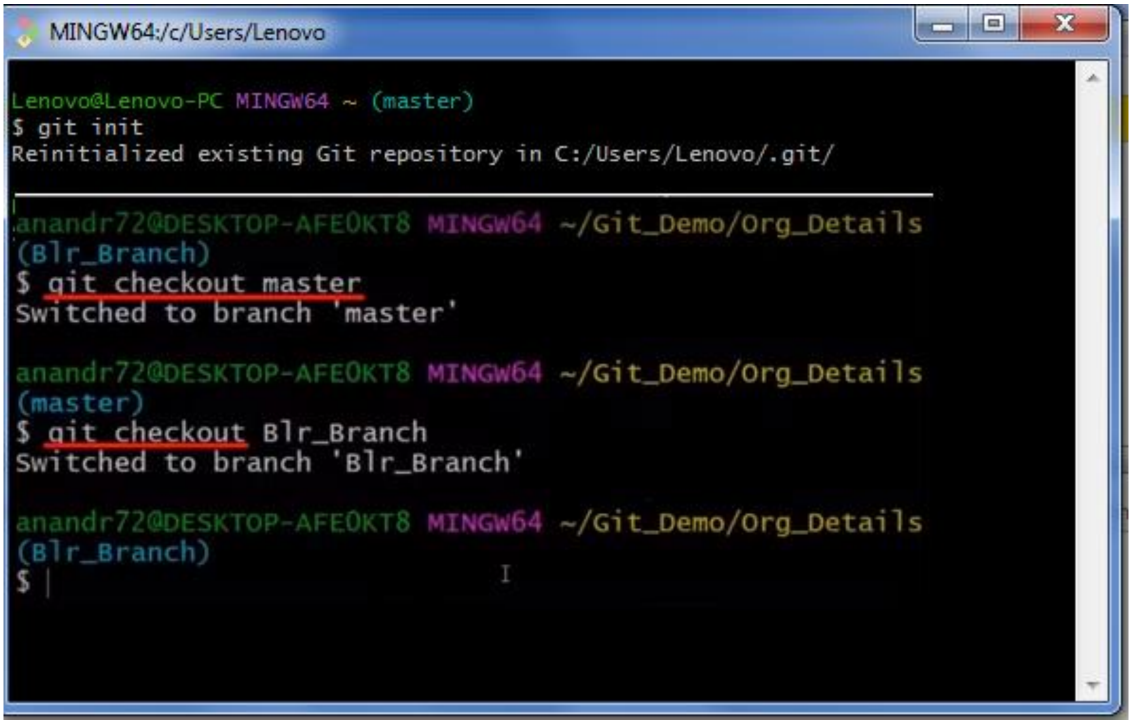
git checkout -b <branch_name>



```
$ git add exp1.doc
fatal: pathspec 'exp1.doc' did not match any files

Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git add download.jpeg
fatal: pathspec 'download.jpeg' did not match any files

Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git push orgin name
error: src refspec name does not match any
error: failed to push some refs to 'orgin'

Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git checkout -b name
Switched to a new branch 'name'

Lenovo@Lenovo-PC MINGW64 ~ (name)
$ git checkout name
error: pathspec 'name' did not match any file(s) known to git

Lenovo@Lenovo-PC MINGW64 ~ (name)
$ git checkout main
error: pathspec 'main' did not match any file(s) known to git
```

8. Switch to a different branch:

git checkout <branch_name>



```
Lenovo@Lenovo-PC MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/Lenovo/.git/

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(Blr_Branch)
$ git checkout master
Switched to branch 'master'

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$ git checkout Blr_Branch
Switched to branch 'Blr_Branch'

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(Blr_Branch)
$
```

These are just a few examples of common Git operations. Git provides a wide range of commands and options to manage version control for your projects.