Git Commands:

- git --version:



  shows the current version of the git which is already installed on the system

- mkdir Git:

  creates a new directory

- cd Git:
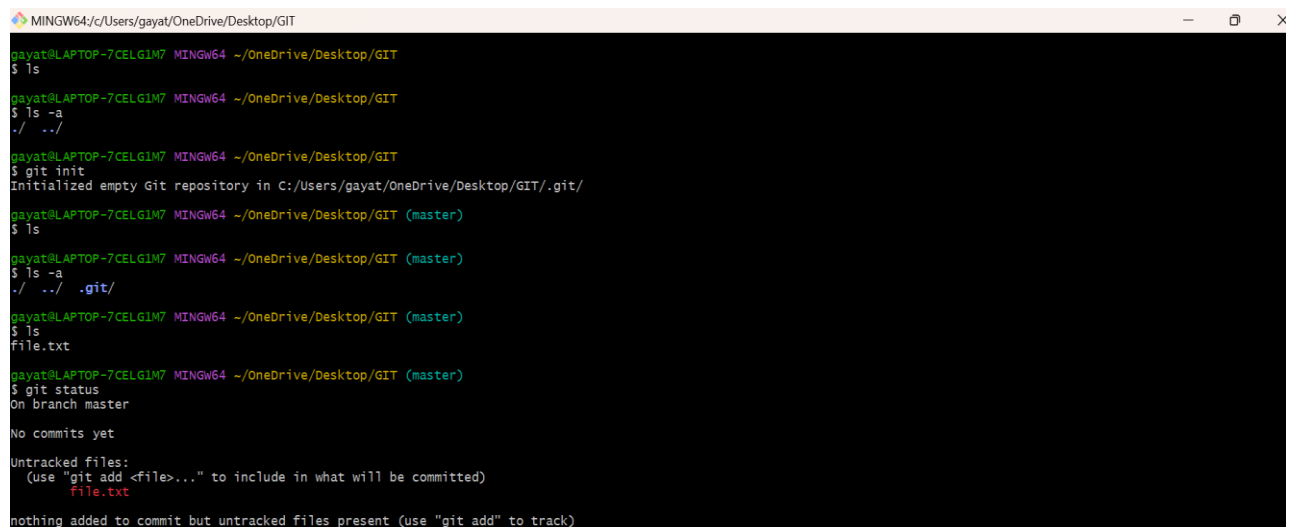  **changes the current working directory**

- git init:
  initialize the empty Git repository in the current directory
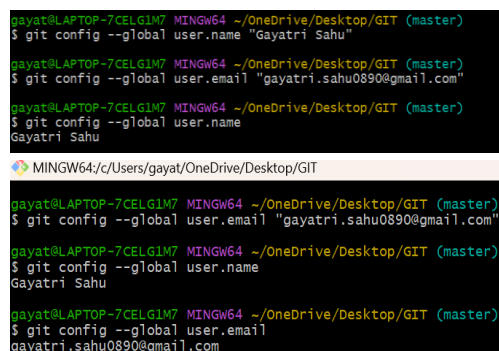
- ls:
  list  the files in the directory

- ls -a:
  list files also hidden files in the directory



  After creating text file , if we write ls command it is showing file.txt

- git config –global user.name "user name":
  set the name of user written in double quotation globally

- git config –global user.email "user email":
  set the email of user written in double quotation globally

- echo hello >> file.txt:

  after creating a text file in the current folder , hello is written by this command in file.txt

- git mv file.txt file1.txt:

  it changes the filename file to file1 without creating a new file

```
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git checkout -f

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git mv file.txt file1.txt

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        renamed:    file.txt -> file1.txt
```

- git add file.txt:

  add file.txt to the staging environment(after finishing part or most of work , we should add the file in staging area). Staging files are files that are ready to be committed to the repository we are working

```
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git add file.txt
warning: in the working copy of 'file.txt', LF will be replaced by CRLF the next
 time Git touches it

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git status
On branch master

No commits yet

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

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git commit -m "file updated"
[master (root-commit) a8a2148] file updated
 1 file changed, 1 insertion(+)
 create mode 100644 file.txt

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git status
On branch master
nothing to commit, working tree clean

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git log
commit a8a2148bdfb95d680def199f56670f21098add1a (HEAD -> master)
Author: Gayatri Sahu <gayatri.sahu0890@gmail.com>
Date:   Wed Apr 2 15:22:26 2025 +0530
```

- git log:

  it shows history of commit of current repository

- git checkout -f:

  it backtracking the file . Discard made the changes made to the specified file and reverse it back to the version in the last commit

- git ls-files:

  only committed files will be displayed in the list

- git add –all:

  add all modified and new files to the staging area without specifying particular filename

- git status:

  show the current state of the repository , including tracked and untracked files , modified files and branch information . In green colour the filename file.txt shown, which is ready to be staged

- git commit -m "message":

  commit keep track of the progress and it is a save point. It performs commit with a message

- git branch 2 times tab click:

  show all the branches present in the directory

- git branch "wipro":

  creates a new branch wipro

- git checkout wipro:

  switches the current branch name to wipro branch

```
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git branch
HEAD      master

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git branch "Wipro"

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git branch
HEAD      Wipro      master

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git checkout Wipro
Switched to branch 'Wipro'

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (Wipro)
$ git status
On branch Wipro
nothing to commit, working tree clean

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (Wipro)
$ git status
On branch Wipro
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file3.txt

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

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (Wipro)
$ git ls-files
file1.txt
file2.txt
```

- git checkout -b wipro:

  it creates a new branch wipro and switches to it

- git merge <branch>:

  it merges the specified branch to the current branch

- git rm file.txt:

  Removes a file from both the working directory and the repository, staging the deletion

- git restore file.txt:

  Restores the file in the working directory to its state in the last commit.

- git clone url:

  to clone github repository locally , we use this command

```
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (Wipro1)
$ git checkout master
Switched to branch 'master'

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ mkdir clone

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ cd clone

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone (master)
$ git clone https://github.com/sahugayatri/wipro.git
Cloning into 'wipro'...
remote: Enumerating objects: 15, done.
remote: Counting objects: 100% (15/15), done.
remote: Compressing objects: 100% (11/11), done.
remote: Total 15 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (15/15), 5.16 KiB | 754.00 KiB/s, done.

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ./

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

```
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone (master)
$ ls
wipro/

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone (master)
$ cd wipro/

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone/wipro (main)
$ ls
README.md  index.html

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone/wipro (main)
$ git ls-files
README.md
index.html

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone/wipro (main)
$ cd ..

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone (master)
$ cd ..

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ mkdir clone1
c
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT (master)
$ cd clone1

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1 (master)
$ git clone1 https://github.com/sahugayatri/wipro.git
git: 'clone1' is not a git command. See 'git --help'.

The most similar command is
        clone

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1 (master)
$ git clone https://github.com/sahugayatri/wipro.git
Cloning into 'wipro'...
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 19 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (19/19), 6.13 KiB | 522.00 KiB/s, done.
```

To delete a file after cloning into a folder :

```
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1 (master)
$ ls
wipro/

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1 (master)
$ cd wipro/

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro (main)
$ git ls-files
2141019146/hello.txt
README.md
index.html

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro (main)
$ cd ..
```

```
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1 (master)
$ ls
wipro/

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1 (master)
$ cd wipro/

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro (main)
$ ls
2141019146/  README.md  index.html

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro (main)
$ cd 2141019146

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ ls
hello.txt

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ ls -a
./  ../  hello.txt

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ rm hello.txt

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git ls-files
hello.txt

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ ls

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ ls -a
./  ../

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    hello.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

```
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    hello.txt

no changes added to commit (use "git add" and/or "git commit -a")

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git commit -m "hello.txt deleted"
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    hello.txt

no changes added to commit (use "git add" and/or "git commit -a")

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ touch file.txt

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ echo wipro >> file.txt

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    hello.txt

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

no changes added to commit (use "git add" and/or "git commit -a")

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git add file.txt
warning: in the working copy of '2141019146/file.txt', LF will be replaced by CRLF the next time Git touch
es it

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git status
```

```
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git commit -m "file is updated"
[main 682b7d9] file is updated
 1 file changed, 1 insertion(+)
 create mode 100644 2141019146/file.txt
```

```
gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git rm file.txt
rm '2141019146/file.txt'

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git ls-files
hello.txt

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git rm hello.txt
rm '2141019146/hello.txt'

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git ls-files

gayat@LAPTOP-7CELG1M7 MINGW64 ~/OneDrive/Desktop/GIT/clone1/wipro/2141019146 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    file.txt
        deleted:    hello.txt
```