

# GIT

- Git is a distributed version control system: tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows.
- Various ways to use git:
  1. The Command Line
  2. Code Editors and IDE
  3. GUI for example GitKraken
- Configuration settings has 3 different levels:
  1. System that applies to all users.
  2. Global that applies to all repository
  3. Local that applies to the current repository.
- At first we need to initialise an empty repository and here the subdirectory will be hidden.
- Git Workflow:
  - ➔ You will have a project directory and to add it on git first you will add it to the staging area and from there you will finally commit to a git repository.
- Commit contains following information:
  - ➔ Id , message, date/time, author, complete snapshots.
- Git compresses the content and doesn't store duplicate content.
- The best practices to commit:
  - ➔ The commit should neither be too big nor too small.
  - ➔ Make meaningful commit messages.
- We can also skip the staging area and can directly commit.

Date: 27-12-2022

Following are the snaps of some of the git commands that I learnt from tutorial:

```
Admin@DESKTOP-27QGE9T MINGW64 ~  
$ git config --global core.autocrlf true  
  
Admin@DESKTOP-27QGE9T MINGW64 ~  
$ mkdir project  
  
Admin@DESKTOP-27QGE9T MINGW64 ~  
$ cd project  
  
Admin@DESKTOP-27QGE9T MINGW64 ~/project  
$ mkdir v  
  
Admin@DESKTOP-27QGE9T MINGW64 ~/project  
$ cd v  
  
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v  
$ git init  
Initialized empty Git repository in C:/Users/Admin/project/v/.git/
```

```
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)  
$ ls  
  
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)  
$ ls -a  
./ ../ .git/  
  
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)  
$ open .git  
bash: open: command not found  
  
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)  
$ echo hello > file1.txt  
  
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)  
$ echo hellow > file2.txt  
  
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)  
$ git status  
On branch master  
  
No commits yet
```

Date: 27-12-2022

```
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git add *.txt
warning: in the working copy of 'file1.txt', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'file2.txt', LF will be replaced by CRLF the next time Git touches it

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   file1.txt
        new file:   file2.txt

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git commit -m "committed"
[master (root-commit) 38a1167] committed
 2 files changed, 2 insertions(+)
 create mode 100644 file1.txt
```

Activat  
Go to Set

```
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git status
On branch master
nothing to commit, working tree clean

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ echo world >> file1.txt

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   file1.txt

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

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git commit -am "skipped staging area"
warning: in the working copy of 'file1.txt', LF will be replaced by CRLF the next time Git touches it
[master fbf9086] skipped staging area
 1 file changed, 1 insertion(+)
```

```
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git add file2.txt

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git ls-files
file1.txt

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git commit -m "fixed"
[master 97caef6] fixed
 1 file changed, 1 deletion(-)
 delete mode 100644 file2.txt

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ mv file1.txt main.js

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git status
On branch master
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:    file1.txt
```

Date: 27-12-2022

```
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git mv main.js file1.js

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        renamed:   file1.txt -> file1.js

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git commit -m "REfactoring"
[master 8cfc0ea] REfactoring
 1 file changed, 0 insertions(+), 0 deletions(-)
 rename file1.txt => file1.js (100%)
```

```
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ mkdir logs

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ echo hello > logs/dev.log

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

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

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

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ echo logs/> .gitignore
```

```
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

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

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

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git commit -m "final commit"
[master ac7dacb] final commit
 1 file changed, 1 insertion(+)
 create mode 100644 .gitignore
```

Date: 27-12-2022

```
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git status -s

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ code .gitignore

Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git status -s
M .gitignore
```

```
Admin@DESKTOP-27QGE9T MINGW64 ~/project/v (master)
$ git log
commit a1a9e34f3597cb1d8ecde4ffe2e27dab9bc4d722 (HEAD -> master)
Author: VidhiTrivedi8 <vidhit@glidemtech.com>
Date: Tue Dec 27 14:44:23 2022 +0530

    done

commit ac7dacb6a9308e862c6b608e628c6275d0339024
Author: VidhiTrivedi8 <vidhit@glidemtech.com>
Date: Tue Dec 27 14:40:18 2022 +0530

    final commit

commit 44941b51503e66e22dc2717b499f22b4e555eeab (origin/master, origin/HEAD)
Author: VidhiTrivedi8 <vidhit@glidemtech.com>
Date: Tue Dec 27 14:34:03 2022 +0530

    Delete file1.js

commit 93706768c50500859771cfb70e426ea3bbb45574
Author: VidhiTrivedi8 <vidhit@glidemtech.com>
```