

# Task1

## 1. git --version

This command is used to check the version of the git.

```
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git --version
git version 2.25.1
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$
```

## 2. git init

This command is used to start new empty git repository

```
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git init
Initialized empty Git repository in /home/varsha/Desktop/ineuron_tasks/assignments/assignment1/.git/
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$
```

## 3. git config

This command is used initially to configure the user.name and user.email. This specifies what email id and username will be used from a local repository.

--global flag is used to writes the settings to all repositories on the computer.

```
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git config --global user.name "VKvarsha98"
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$

varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git config --global user.email "varshavk98@gmail.com"
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$
```

## 4. git status

The git status command tells the current state of the repository.

```
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Task1/screenshots/gitadd.png
    Task1/screenshots/gitbranch.png
    Task1/screenshots/gitcommit.png
    Task1/screenshots/gitstatus.png

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

## 5. git add

git add . adds whole files to the staging area.

```
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git add .
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$
```

## 6. git commit

This command ensures that the changes are saved to the local repository.

“-m <message>” is used to describe everyone that what has happened for easy understanding.

```

varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git commit -m "first commit"
[master (root-commit) 6bdb60b] first commit
6 files changed, 2 insertions(+)
create mode 100644 README.md
create mode 100644 Task1/screenshots/gitconfigemail.png
create mode 100644 Task1/screenshots/gitconfigname.png
create mode 100644 Task1/screenshots/githelp.png
create mode 100644 Task1/screenshots/gitinit.png
create mode 100644 Task1/screenshots/gitversion.png
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$

```

## 7. git branch

The git branch command is used to determine what branch the local repository is on.

```

varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git branch -M main
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$

```

git branch <branch name> creates new branch

```

varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git branch developer1
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git branch
  developer1
* main

```

## 8. git remote

This command used to create, view, and delete connections to other repositories.

Create a new connection to a remote repository

```

varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git remote add origin https://github.com/VKvarsha98/Git_Assignment.git

```

List the remote connections

```

varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git remote -v
origin https://github.com/VKvarsha98/Git_Assignment.git (fetch)
origin https://github.com/VKvarsha98/Git_Assignment.git (push)

```

## 9. git push

This command is used to transfer the commits or pushing the content from the local repository to the remote repository.

```

varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git push -u origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (10/10), 90.55 KiB | 10.06 MiB/s, done.
Total 10 (delta 0), reused 0 (delta 0)
To https://github.com/VKvarsha98/Git_Assignment.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$

```

## 10. git checkout

The git checkout command is used to switch branches, whenever the work is to be started on a different branch.

```

● varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git checkout developer1
Switched to branch 'developer1'
● varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git branch
* developer1
  main
○ varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$

```

## 11.git merge

This command is used to integrate the branches together. The command combines the changes from one branch to another branch.

```

varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git merge developer1
Updating 1a9f900..e8af0c5
Fast-forward
 README.md | 4 ++--
 Task1/screenshots/gitadd.png | Bin 0 -> 10448 bytes
 .../screenshots/gitbranch.png | Bin 0 -> 10163 bytes
 .../gitbranchdev.png | Bin 0 -> 13825 bytes
 .../gitcheckout.png | Bin 0 -> 20671 bytes
 .../screenshots/gitcommit.png | Bin 0 -> 35649 bytes
 Task1/screenshots/gitlog.png | Bin 0 -> 18299 bytes
 Task1/screenshots/gitmerge.png | Bin 0 -> 19626 bytes
 Task1/screenshots/gitpush.png | Bin 0 -> 36824 bytes
 .../gitremote-v.png | Bin 0 -> 14495 bytes
 .../gitremoteadd.png | Bin 0 -> 9921 bytes
 .../screenshots/gitstatus.png | Bin 0 -> 25784 bytes
 .../gitwhatchanged.png | Bin 0 -> 47124 bytes
13 files changed, 2 insertions(+), 2 deletions(-)
create mode 100644 Task1/screenshots/gitadd.png
create mode 100644 Task1/screenshots/gitbranch.png
create mode 100644 Task1/screenshots/gitbranchdev.png
create mode 100644 Task1/screenshots/gitcheckout.png
create mode 100644 Task1/screenshots/gitcommit.png
create mode 100644 Task1/screenshots/gitlog.png
create mode 100644 Task1/screenshots/gitmerge.png
create mode 100644 Task1/screenshots/gitpush.png
create mode 100644 Task1/screenshots/gitremote-v.png
create mode 100644 Task1/screenshots/gitremoteadd.png
create mode 100644 Task1/screenshots/gitstatus.png
create mode 100644 Task1/screenshots/gitwhatchanged.png

```

## 12.git log

This command shows the order of the commit history for a repository.

```

● varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git log
commit 6bdb60bce0ae29c109558f096ac5c7c422070f5c (HEAD -> master)
Author: VKvarsha98 <varshavk98@gmail.com>
Date: Thu Oct 13 15:56:14 2022 +0530

    first commit

```

## 13.git pull

This command is used to fetch and merge changes from the remote repository to the local repository.

```
● varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git pull
Already up to date.
○ varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$
```

## 14.git reset

This command is used to unstage a file while retaining the changes in working directory

```
● varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git reset 83a473ea39518b4585c644e8d492a6c018c59130
Unstaged changes after reset:
M   main.py
```

## 15.git stash

This command is used to save modified and staged changes.

git stash pop command is used to restore working directory from top of the stash stack

```
● varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git stash
Saved working directory and index state WIP on main: 83a473e added main.py
● varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git stash pop
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)
    new file:   Task1/screenshots/gitreset.png

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:   main.py

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Task1/screenshots/gitstash.png

Dropped refs/stash@{0} (ad21111b703cfd2c172405f83415e413c342ae17)
○ varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$
```

## 16.git rm

This command deletes the file from your working directory and stages the deletion.

```
● varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$ git rm -f main.py
rm 'main.py'
○ varsha@varsha:~/Desktop/ineuron_tasks/assignments/assignment1$
```

## 17.git help

This command displays help information about git

```

varsha@varsha:~/Desktop/neuron_tasks/assignments/assignment1$ git help
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
      [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
      [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
      [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
      <command> [<args>]

These are common Git commands used in various situations:


start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one


work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  restore    Restore working tree files
  rm         Remove files from the working tree and from the index
  sparse-checkout Initialize and modify the sparse-checkout


examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  diff       Show changes between commits, commit and working tree, etc
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status


grow, mark and tweak your common history
  branch     List, create, or delete branches
  commit     Record changes to the repository
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  reset      Reset current HEAD to the specified state
  switch     Switch branches
  tag        Create, list, delete or verify a tag object signed with GPG


collaborate (see also: git help workflows)
  fetch      Download objects and refs from another repository
  pull       Fetch from and integrate with another repository or a local branch
  push       Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.

```

## 18.git whatchanged

This command shows logs with difference each commit introduces

```

varsha@varsha:~/Desktop/neuron_tasks/assignments/assignment1$ git whatchanged
commit 6bdb60bce0ae29c109558f096ac5c7c422070f5c (HEAD -> developer1, origin/main, main)
Author: VKvarsha98 <varshavk98@gmail.com>
Date: Thu Oct 13 15:56:14 2022 +0530

    first commit

:000000 100644 0000000 60a1143 A      README.md
:000000 100644 0000000 da8d267 A      Task1/screenshots/gitconfigemail.png
:000000 100644 0000000 525ee95 A      Task1/screenshots/gitconfigname.png
:000000 100644 0000000 7da97ac A      Task1/screenshots/githelp.png
:000000 100644 0000000 ca77194 A      Task1/screenshots/gitinit.png
:000000 100644 0000000 fb12eb9 A      Task1/screenshots/gitversion.png
varsha@varsha:~/Desktop/neuron_tasks/assignments/assignment1$

```

## 19.git clone

This command is used to obtain a repository from an existing URL.

```

varsha@varsha:~/Desktop/neuron_tasks/assignments$ git clone https://github.com/varshavk98/fake-news-detection-using-deep-learning
Cloning into 'fake-news-detection-using-deep-learning'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 2.13 MiB | 2.80 MiB/s, done.
varsha@varsha:~/Desktop/neuron_tasks/assignments$

```