Git

**Version Control System** - It is developed to co-ordinate the work among the developers.

# Features of GIT

**Open Source** – GPL license

**Scalable**- large number of users git can easily handle

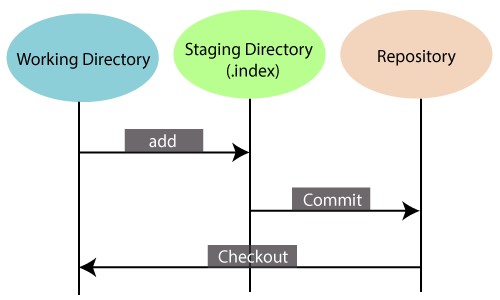
**Distributed**- on another machine user can easily clone

**Security**-Secure, uses SHA1 (Secure Hash Function) to name and identify the objects

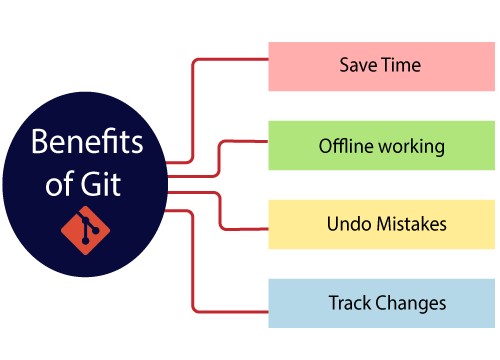
**Speed**-fast, most of the operation on local repo

**Branching** and Merging- great feature, multiple branches so that other developer work together.

**Staging** **Area**- preview of next commit.



# Benefits of using GIT



Installing GIT on Windows <https://git-scm.com/downloads>download and install for this website $ git --version

Register user with git

git config --global user.name "Sushant"

git config --global user.email "Sushant82@gmail.com"

user is successfully registered

git config --list

# Important Terminology

**Branch**- repository diverges from main working directory.

**Checkout**- checkout is used for the act of switching between different versions of a target entity **Clone**: making copy from server.

**Merge** – combining branches

**Origin**- remote repository from a project was initially cloned

**Pull-** receive the data from Server (GITHUB)

**Push-** Upload local repository to sever.

**Git Ignore**-use for intentionally untrack the file

**Git Diff**- shows changes between commit, working tree etc.

**Git Rm**- for removing files.

Etc.

# Let start

Create a local repository:

$ git init

Make copy

$ git clone

Adding file to staging area

$ git add file //single file

$ git add -A //all files

See the status of file

$ git status

Committing the change

## $ git commit -m “comment”

Record the file permanently

Track the changes that have not been staged

$git diff

Track the changes that have staged but not committed

$git diff –-staged

Track the changes after committing a file:

$git diff HEAD

Show the objects

## $ git show

Commit History

Display the most recent commits and status of the head.

$git log $git log -p -2

Output as one commit per line

$git log –-oneline

Display the files that have been modified

$git log –-stat

Display the modification on each line of a file:

## $ git blame <file name>

Ignoring Files

Create. gitignore file

Branching

List a branch

$git branch –-list

Create Branch

$git branch [name]

Delete Branch

$git branch -d [name]

Renaming the branch

## $git branch -m [old name] [new name]

Git checkout

Switch between branch in a repository

$git checkout [branch name]

Create new branch and switch to it

## $git checkout -b [branch name]

Merging

Merge the branches

$git merge [branch name]

Working on Remote

$git remote -v

Add remote to repository

## $git remote add [name] [remote url]

Remove from

Delete the file

$git rm [file]

Only remove file from staging area

$git rm –-cached [file]

# GITHUB

Repository Hosting Service Remote Repository:

Git remote add name url

Git remote -v

Git push -u origin master

Git remote set-url origin url