GIT

GIT

Git is a free, open source distributed version control system tool designed to handle everything from small to very large projects with speed and efficiency.

GIT

In Distributed VCS, **every contributor has a local copy** or “clone” of the main repository i.e. everyone maintains a local repository of their own which contains all the files and metadata present in the main repository.

Contributors update their local repositories with new data from the central server by an operation called “**pull**” and affect changes to the main repository by an operation called “**push**” from their local repository.

GIT

A few Operations & Commands

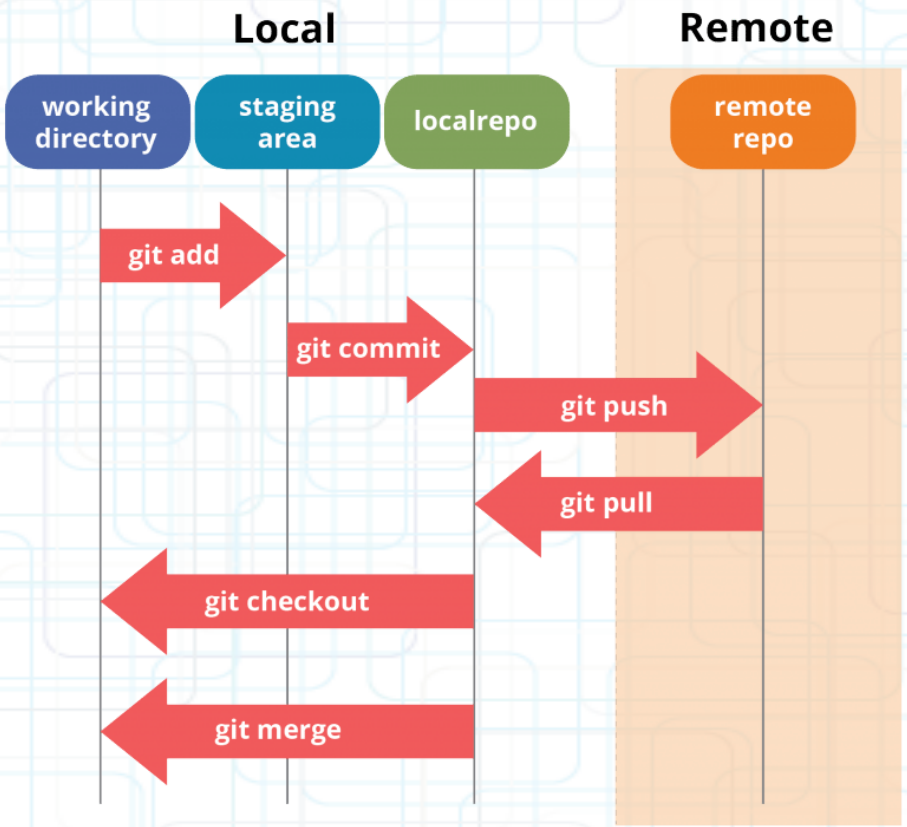
Some of the basic operations in Git are:

1.Initialize
2.Add
3.Commit
4.Pull
5.Push
6.Status
7.log

Some advanced Git operations are:

1.Branching
2.Merging
3.Rebasing

GIT


Local
Remote
working directory
staging area
localrepo
remote repo
git add
git commit
git push
git pull
git checkout
git merge

GIT

Working with an existing repositry

Create a new folder or in an clean existing folder, issue the following clone command

git clone https://github.com/abababababab/training.git

where “https://github.com/abababababab/training.git” is the repo url.

Once the command execution completes, verify contents of folder(s) created (in our case the training and its subfolders)

GIT

Adding a new file or a new folder

git add <directory>
or
git add <file>

Verify that your new file or folder that you added is “staged” by issuing the status command.
git status

```
c:\Users\home\Training2\training\Day2>git add GIT.pdf

c:\Users\home\Training2\training\Day2>git status
On branch main
Your branch is up-to-date with 'origin/main'.
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   GIT.pdf


c:\Users\home\Training2\training\Day2>
```

GIT

Commit the changes to Staging area

git commit
This will commit the staged snapshot and will launch a text editor prompting you for a commit message.

Or you can use:
git commit -m “<message>”
Let’s try it out.

```
c:\Users\home\Training\Day1>git commit -m "Day 1 commit While Do-while For"
[main 5457572] Day 1 commit While Do-while For
 3 files changed, 41 insertions(+)
 create mode 100644 Day1/TestDWhile.java
 create mode 100644 Day1/TestDoWhile.java
 create mode 100644 Day1/TestPrime.java
```

GIT

Push the changes to Remote Repo

If you have the permissions, you can commit the changes to the remote repo.

Use the **push** command as below:
c:\Users\home\Training\Day1>git push -u origin main

where “main” is the name of the branch you want to push your changes.

```
c:\Users\home\Training2\training\Day2>git push
Fatal: HttpRequestException encountered.
Username for 'https://github.com': [redacted]
Password for 'https://[redacted]@github.com':
Counting objects: 4, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 444.21 KiB | 0 bytes/s, done.
Total 4 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/[redacted]/training.git
   24e0703..48b92e4  main -> main

c:\Users\home\Training2\training\Day2>
```

GIT

Keeping your local repo Up-to-date

It is a good practice to frequently update your local workspace(repo) in sync with the remote repo.

Use the **pull** command as below:
c:\Users\home\Training\Day1>git pull

Git will automatically get the latest contents from the remote repo and update your local repo contents.

In case there is any conflict with the contents that you have modified (in your local remote), it will let you know.

If there nothing new in the remote repo, you will see a message like this

```
c:\Users\home\Training2\training\Day2>git pull
Already up-to-date.
```

GIT

Knowing who commited what

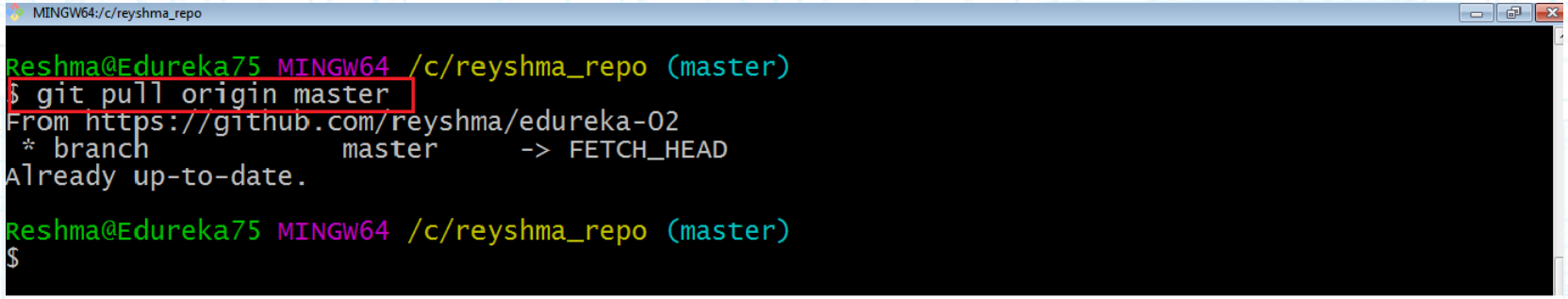
The **git log** command will list out the various commits that went into the repos.

```
c:\Users\home\Training2\training\Day2>git log
commit 24e070355a3699556600b314e02d8997eb9ee454
Author: Your Name <you@example.com>
Date:   Fri Oct 8 15:13:09 2021 +0530

    oops concepts

commit 4fcf448fd3628885280952422fc88e16cd5fbb34
Author: Your Name <you@example.com>
Date:   Fri Oct 8 09:42:22 2021 +0530

    testing add and commit and push

commit 5457572a3b81d4ea12d7725896345acbf72286c2
Author: Your Name <you@example.com>
Date:   Fri Oct 8 07:59:25 2021 +0530
```

GIT

git pull origin master

This command will copy all the files from the master branch of remote repository to your local repository.



Since my local repository was already updated with files from master branch, hence the message is Already up-to-date. Refer to the screen shot above.

Note: *One can also try pulling files from a different branch using the following command:*
git pull origin <branch-name>

Your local Git repository is now updated with all the recent changes. It is time you make changes in the central repository by using the **push** command.