

## GIT

- Distributed Version Control.
- Track history, work together.

### Three States,


- Working directory.
- Staging area.
- Repository (.git directory).

## Commit

### Commands,

git init	- initialize an empty git repository.
git status	- status of our working directory and staging area.
git add<file>	- add a file to the staging area.
git commit -m "message here"	- commit a file to our git repository (.git directory).
git log	- history of our commits (snapshots).

## Quiz completion certificate

 Intro to Git ★ Leave a review

**Great job! You are ready to move on to the next lecture.**  
You got 6 out of 6 correct.

✓ **What you know** ⓘ

Which command do we use to initialize an empty git repository?

Which command do we use to add a file to the staging area?

Which command do we use to commit a file to our git repository (.git directory)?

Which command do we use to check the status of our working directory and staging area?

What are the three states of git?

Which command do we use to check the history of our commits (snapshots)?

Fullscreen

Retry quiz Continue ⚙️ ↶ ↷

## Adding, Removing and Ignoring

### Commands,

git add -A/ --all        - add all files and folders (including hidden ones) to the staging area.  
git reset HEAD <filename> - remove file from the staging area.  
git add\*.html            - add all files of the same type (e.g., all **.html** files).  
.gitignore               - list all of the files and folders we want git to ignore.

## Quiz completion certificate

**Great job! You are ready to move on to the next lecture.**

You got 4 out of 4 correct.

✓ **What you know** ⓘ

Which command do we use to add all files and folders (including hidden ones) to the staging area?

Which command do we use to add all files of the same type?(e.g., all .html files)

Which command do we use to remove helloWorld.js from the staging area?

What is the name of the file that we create in order to list all of the files and folders we want git to ign...

Fullscreen


Retry quiz Continue ⚙ ↶ ↷

## Branches

### Commands,

git branch	- list our branches.
git checkout -b <branch_name>	- add and checkout a new branch, simultaneously.
git checkout <branch_name>	- switch to another branch.
git merge <branch_name>	- merge the branches.
git branch-d <branch_name>	- remove an unwanted branch.

## Quiz completion certificate

Intro to Git★ Leave a review

**Great job! You are ready to move on to the next lecture.**  
You got 5 out of 5 correct.

✓ What you know ⓘ

Which command do we use to list our branches?

Which command do we use to add and checkout a new branch, simultaneously?




Which command do we use to switch to another branch?

What are the steps for merging a feature branch into master?


Which command do we use to remove an unwanted branch?

Retry quiz

Continue



## Course completion certificate

Intro to Git★ Leave a review

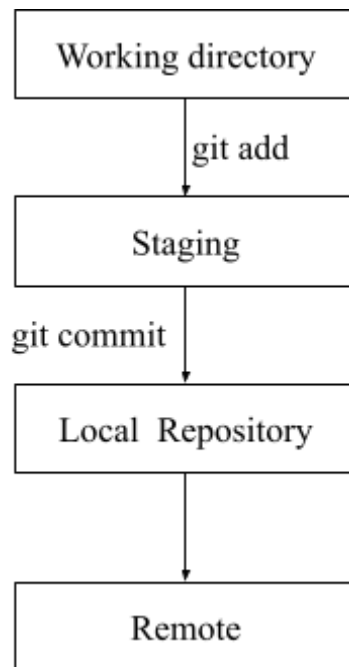
Well done!

You've successfully completed the Intro to Git course.

If you enjoyed this course and are looking for what to learn next, then head over to [DevSprout.io](https://devsprout.io) to keep up to date with my latest work.

Git has two environments,

- Local - Only the system user can access it.
- Remote - Share a repository.



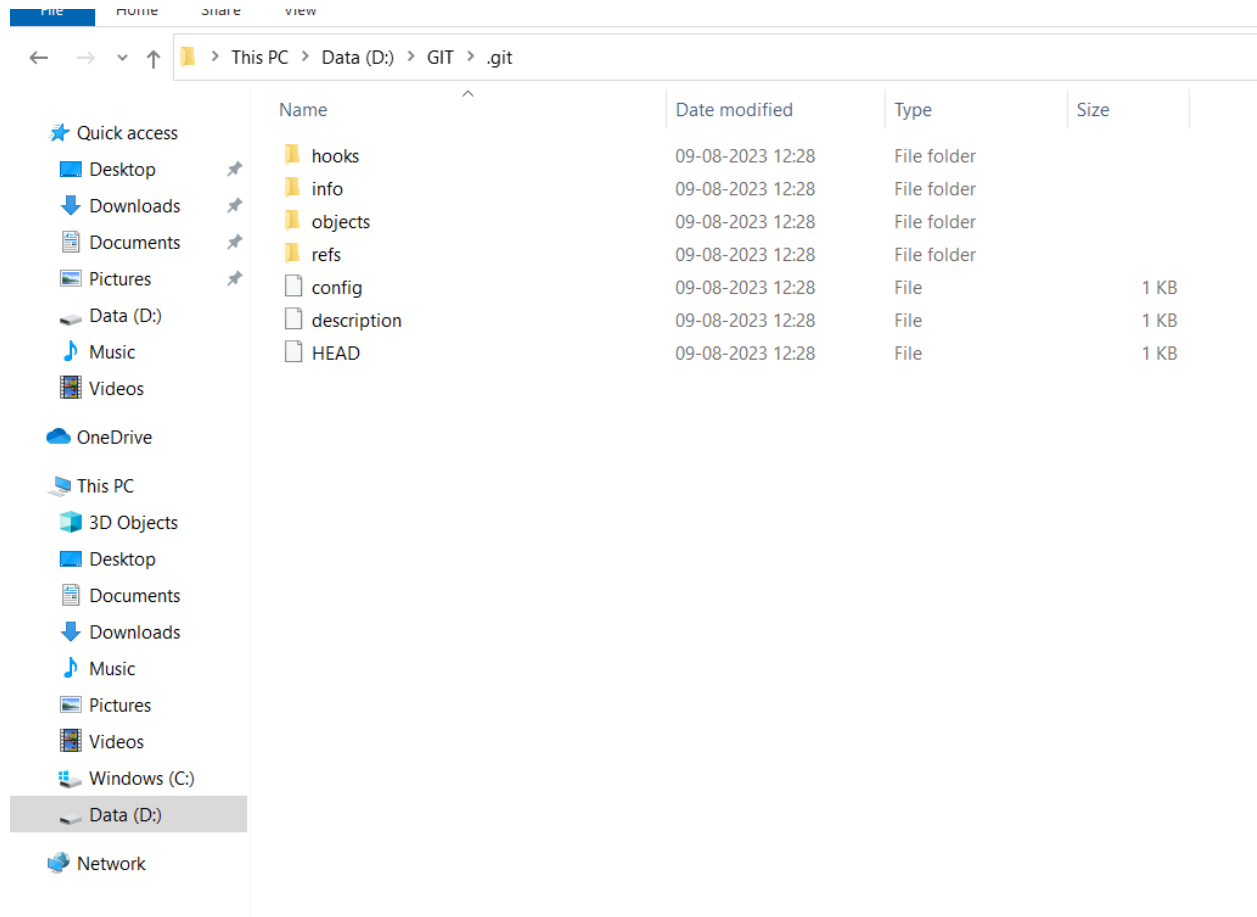
MINGW64:/d/GIT

```
nagajeyandhan.ravi@RLP1638 MINGW64 ~  
$ pwd  
/c/Users/nagajeyandhan.ravi  
nagajeyandhan.ravi@RLP1638 MINGW64 ~  
$ cd D:\GIT
```

**git-init** - initialize an empty git repository

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT  
$ git init  
Initialized empty Git repository in D:/GIT/.git/
```

## Git local repository



**git status** - status of our working directory and staging area

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git status
On branch master

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$
```

**Adding to the stage** - add a file to the staging area

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git add -A

nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git status
On branch master

No commits yet

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

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ touch three.txt

nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git status
On branch master

No commits yet

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

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:   one.txt.txt
    modified:   test/Two.txt.txt

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

**Modified** - If any changes occur.

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git status
On branch master

No commits yet

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

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:   one.txt.txt
        modified:   test/Two.txt.txt
```

**List** - List all the files.

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ ls
one.txt  test/  three.txt
```

**Reset** - Remove all the files from the staging area.

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git reset

nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git status
On branch master

No commits yet

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

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

## Commit

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git commit -m "Added 1 file"
[master (root-commit) ed6019b] Added 1 file
Committer: Nagajeyandhan Ravi <nagajeyandhan.ravi@itp.objectfrontier.com>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:
```

```
git config --global --edit
```

After doing this, you may fix the identity used for this commit with:

```
git commit --amend --reset-author
```

```
3 files changed, 2 insertions(+)
create mode 100644 one.txt
create mode 100644 test/two.txt
create mode 100644 three.txt
```

**Log** - history of our commits (snapshots)

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git log
commit ed6019b3386b26a37f9fa4df337a0f5e6d9bec33 (HEAD -> master)
Author: Nagajeyandhan Ravi <nagajeyandhan.ravi@itp.objectfrontier.com>
Date: Wed Aug 9 16:39:21 2023 +0530

    Added 1 file
```



**Diff** - To view the modified changes

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (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:   one.txt

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

nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git diff
diff --git a/one.txt b/one.txt
index 71681a7..8c705ca 100644
--- a/one.txt
+++ b/one.txt
@@ -1,1 @@
-Add file One.txt
\ No newline at end of file
+Add file and it is the first one.
\ No newline at end of file
```

**Diff -cached** - To View the changes in the file by its name

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git diff --cached one.txt
diff --git a/one.txt b/one.txt
index 71681a7..8c705ca 100644
--- a/one.txt
+++ b/one.txt
@@ -1,1 @@
-Add file One.txt
\ No newline at end of file
+Add file and it is the first one.
\ No newline at end of file
```

**git rm** - delete a particular file.

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git rm three.txt
rm 'three.txt'

nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   one.txt
        deleted:    three.txt
```

**git rm -f**

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git rm -f one.txt
rm 'one.txt'

nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    one.txt
        deleted:    three.txt
```


**git reset --hard** - reset all the changes done.

```
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git reset --hard
HEAD is now at ed6019b Added 1 file

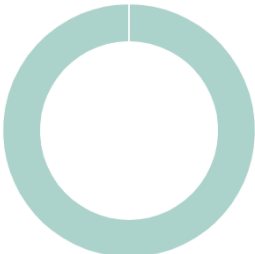
nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ git status
On branch master
nothing to commit, working tree clean

nagajeyandhan.ravi@RLP1638 MINGW64 /d/GIT (master)
$ ls
one.txt  test/  three.txt
```

## Quiz completion certificate

 Git tutorials for Beginners - Complete Course ★ Leave a review

7 questions | 15 minutes | 80% correct required to pass



Attempt 2: **Passed!** (80% required to pass) ^

**100%** correct (7/7)

42 seconds

August 9, 2023 at 05:15 PM

Review questions

Correct

Wrong

Skipped/Unanswered

Retake test

Continue

