# **GIT**

## **Definition:**

Git is a version control system, that helps in track changes is files, allowing multiple developers to work on same project without overwriting each other codes. it is also used for managing different version of source code. It also enables branching, merging, and maintaining a history of changes.

## **Basic Commands:**

## **Check Git Version:**

### **git --version:**

this command shows the version of git



## **Initialize A Repo:**

### **git init:**

Create an empty Git repository or reinitialize an existing one. And also create a .git file.



## **Remote Repos:**

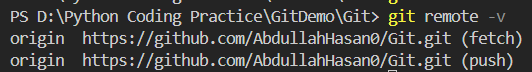
### **git remote add origin <link of repo on GitHub>:**

This command link local repo to our remote repo (on GitHub) allowing push and pull from remote repo



### **git remote -v:**

to verify remote repo



## **Remote Repos Status:**

### **git status:**

Show the working tree status

This command can give 4 possible outputs

Untracked

This is status is shown when git didn’t know a file exist or not, simple when we create a new file on local repo



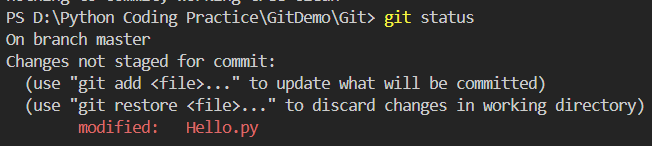
Here I created a Hello.py, and ‘U’ opposite the file name shows that the file in untracked

Modified

When a file changed but not staged for next commit

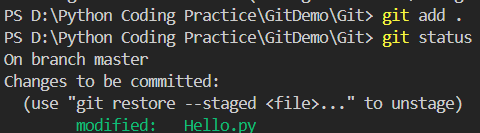


Here I modified Hello.py, and ‘M’ opposite the file name shows that the file in Modified. We can also use the git status to see this.



Staged

When a file is stagged after using **git add** and ready to commit



Here green modified shows that Hello.py is in staging phase

Unmodified

There is no output for this. This means there is not a file that is modified.

## **Staging Changes:**

### **git add:**

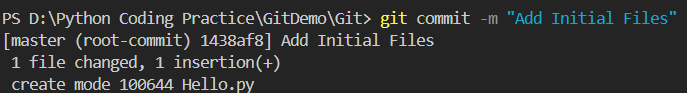
adds new or changed files in your working directory to the git staging area. Staged files are files that are ready to be committed to the repository you are working on.



## **Committing Changes:**

### **git commit -m “Write message here”:**

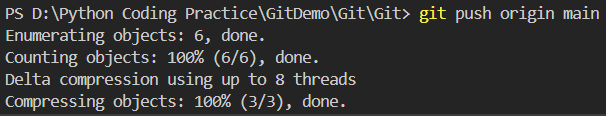
It is the record of change. Adding commits keep track of our progress and changes as we work. Git considers each commit change point or "save point".



## **Pushing Changes:**

### **git push origin main:**

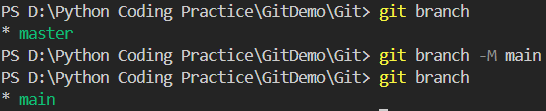
This command uses to push changes from local repo ‘main’ branch to remote repo ‘main’ branch



## **Renaming Branches:**

### **git branch -M ‘Name’:**

This command uses to change the name of branch



## **Cloning Repo:**

### **git clone <link of repo on GitHub>:**

This command uses to create a copy of already existing GitHub repo

