Git Installation

Git is a version control system and is used by most developers nowadays. It allows you to keep track of your code changes, revert to previous stages, create branches, and to collaborate with your fellow developers.

Git is originally developed by Linus Torvalds, the creator of the Linux kernel.

This tutorial will guide you through the steps required to install Git on Ubuntu 18.04. The same instructions apply for Ubuntu 16.04 and any other Ubuntu-based distribution, Linux Mint and Elementary OS.

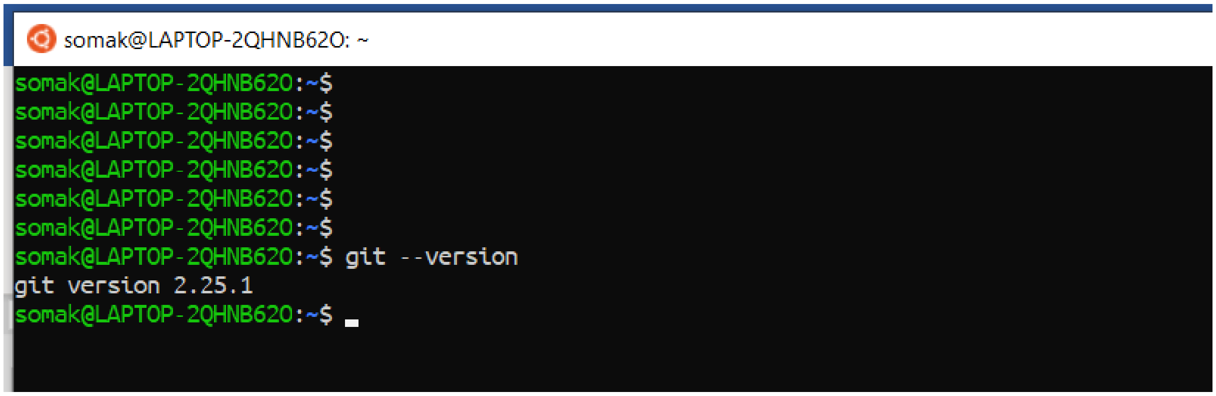
Pre-Requisites

- A system running Ubuntu 18.04
- A user account with sudo privileges
- Access to a terminal window/command-line (Ctrl+Alt+T)

Before continuing with this tutorial, make sure you are logged in as root or a user with sudo privileges.

First open Ubuntu App from Windows 10 or Command prompt from Ubuntu OS. Then check your version of Git by entering the following:

```
$ git --version
```



```
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$ git --version
git version 2.25.1
somak@LAPTOP-2QHNB620:~$
```

Though Git latest version comes with Ubuntu Installation, here we will show if you need to install it from the scratch.

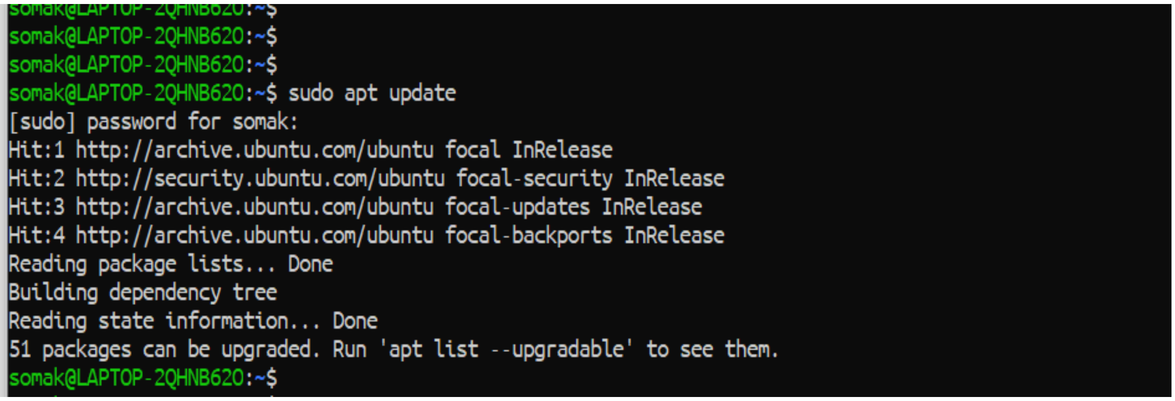
There are 2 way you can install Git:

1. Install using Command Prompt
2. Install using shell scripts

1. Install using Command Prompt

Before installing Git, first we need to run update and upgrade command which helps to update all the necessary parts for Ubuntu. First, run the update command as follow:

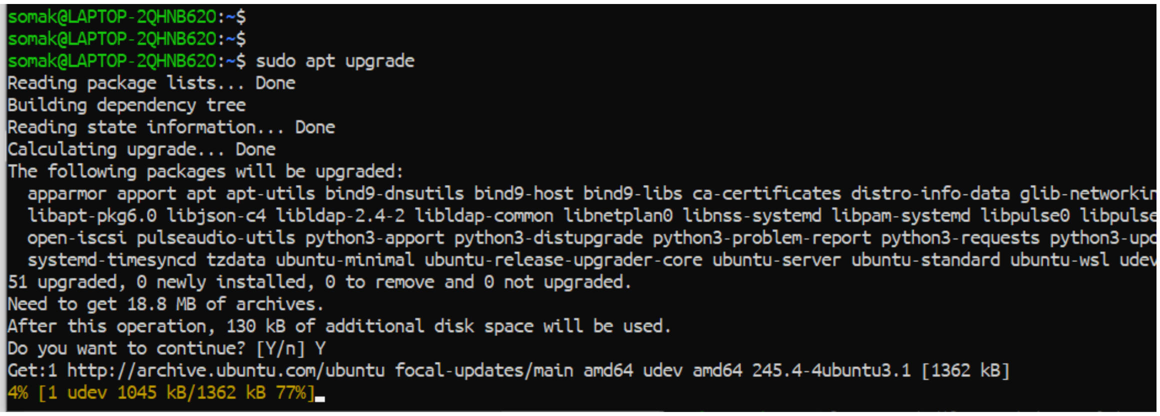
```
$ sudo apt update
```



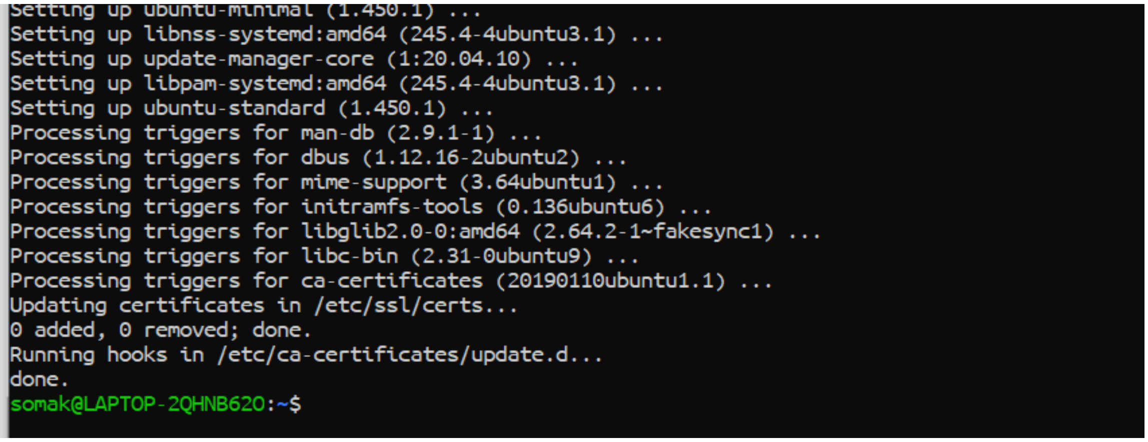
```
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$ sudo apt update
[sudo] password for somak:
Hit:1 http://archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:3 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:4 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
51 packages can be upgraded. Run 'apt list --upgradable' to see them.
somak@LAPTOP-2QHNB620:~$
```

Then run upgrade command as follow:

```
$ sudo apt upgrade
```



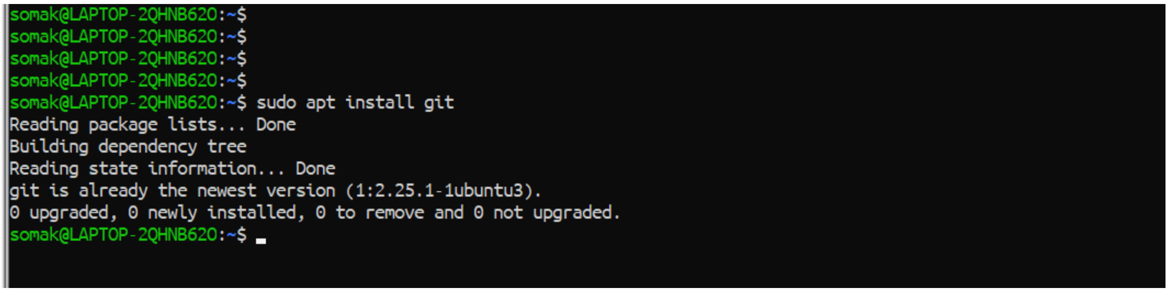
```
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  apparmor apport apt apt-utils bind9-dnsutils bind9-host bind9-libs ca-certificates distro-info-data glib-networkin
  libapt-pkg6.0 libjson-c4 libldap-2.4-2 libldap-common libnetplan0 libnss-systemd libpam-systemd libpulse0 libpulse
  open-iscsi pulseaudio-utils python3-apport python3-distupgrade python3-problem-report python3-requests python3-upd
  systemd-timesyncd tzdata ubuntu-minimal ubuntu-release-upgrader-core ubuntu-server ubuntu-standard ubuntu-wsl udev
51 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 18.8 MB of archives.
After this operation, 130 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 udev amd64 245.4-4ubuntu3.1 [1362 kB]
4% [1 udev 1045 kB/1362 kB 77%]
```



```
Setting up ubuntu-minimal (1.450.1) ...
Setting up libnss-systemd:amd64 (245.4-4ubuntu3.1) ...
Setting up update-manager-core (1:20.04.10) ...
Setting up libpam-systemd:amd64 (245.4-4ubuntu3.1) ...
Setting up ubuntu-standard (1.450.1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for dbus (1.12.16-2ubuntu2) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for initramfs-tools (0.136ubuntu6) ...
Processing triggers for libglib2.0-0:amd64 (2.64.2-1~fakesync1) ...
Processing triggers for libc-bin (2.31-0ubuntu9) ...
Processing triggers for ca-certificates (20190110ubuntu1.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
somak@LAPTOP-2QHNB620:~$
```

Once upgrade is done, we will start to install git as follow:

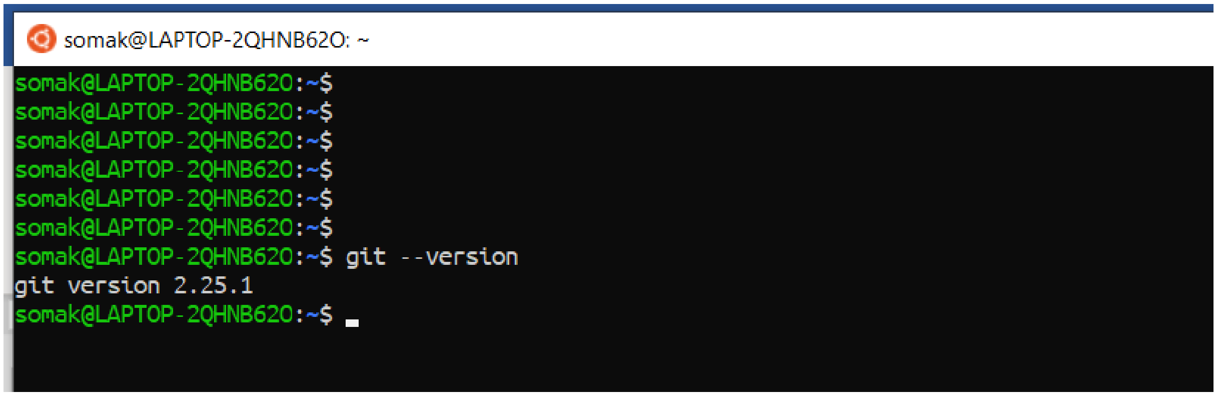
```
$ sudo apt install git
```



```
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$ sudo apt install git
Reading package lists... Done
Building dependency tree
Reading state information... Done
git is already the newest version (1:2.25.1-1ubuntu3).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
somak@LAPTOP-2QHNB620:~$
```

If you installed Git earlier with latest version, then you will see "git is already the newest version" message. If not, then this command will install Git for you. Later you can check the version again.

```
$ git --version
```



somak@LAPTOP-2QHNB620: ~

```
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$ git --version
git version 2.25.1
somak@LAPTOP-2QHNB620:~$
```

2. Install using shell scripts

Now first we need to download the Robotic-Greeter folder from the GitHub link: https://github.com/ripanmukherjee/Robotic-Greeter

You can download it in two ways:

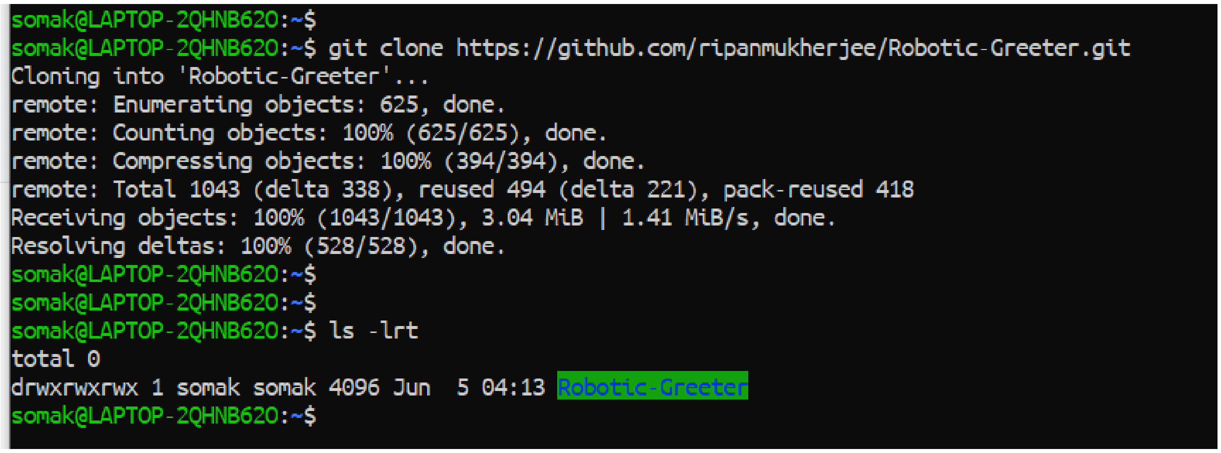
1. Clone it with command prompt
2. Download it as a Zip file

Inside of the Robotic-Greeter folder we have the shell (Unix) script which you need to run, and this script will automatically install Git on your computer. Also, you can keep the shell script for your future reference and whenever you need to update Git version, you can directly run the shell script.

1. Using Clone method

Go to Ubuntu App from Windows 10 or Command prompt from Ubuntu OS and run the following command:

```
$ git clone https://github.com/ripanmukherjee/Robotic-Greeter.git
```

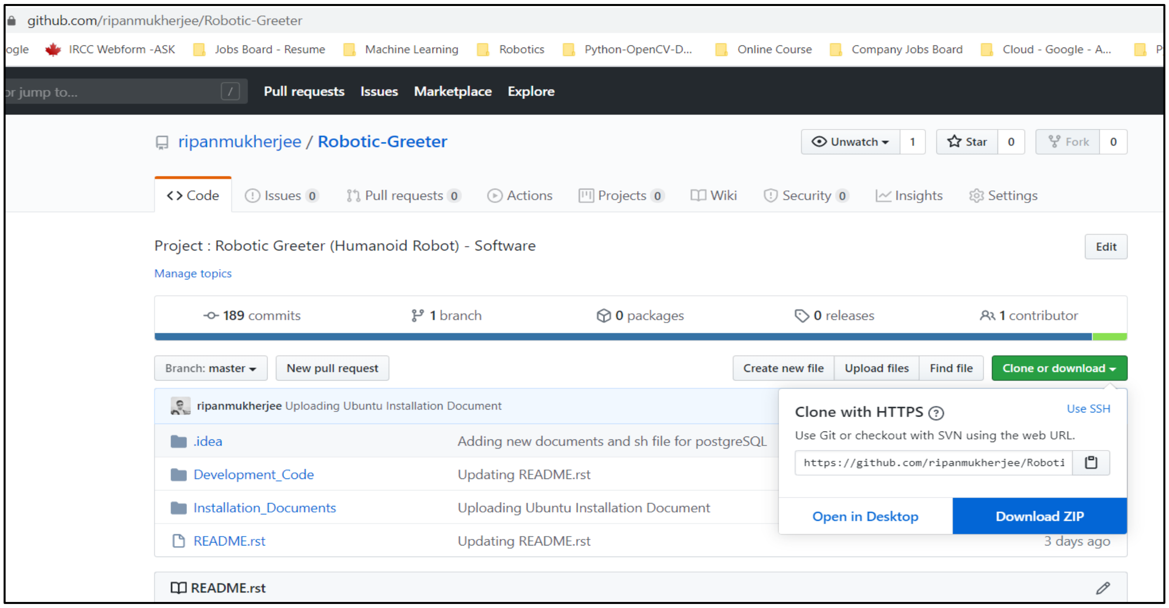


```
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$ git clone https://github.com/ripanmukherjee/Robotic-Greeter.git
Cloning into 'Robotic-Greeter'...
remote: Enumerating objects: 625, done.
remote: Counting objects: 100% (625/625), done.
remote: Compressing objects: 100% (394/394), done.
remote: Total 1043 (delta 338), reused 494 (delta 221), pack-reused 418
Receiving objects: 100% (1043/1043), 3.04 MiB | 1.41 MiB/s, done.
Resolving deltas: 100% (528/528), done.
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$ ls -lrt
total 0
drwxrwxrwx 1 somak somak 4096 Jun  5 04:13 Robotic-Greeter
somak@LAPTOP-2QHNB620:~$
```

This command will automatically download Robotic-Greeter folder on your computer.

2. Download as Zip

Also, you can directly download the Zip file and unzip it. Then you need to put it in proper directory or your project directory.



Project : Robotic Greeter (Humanoid Robot) - Software

ripanmukherjee Uploading Ubuntu Installation Document

.idea	Adding new documents and sh file for postgreSQL
Development_Code	Updating README.rst
Installation_Documents	Uploading Ubuntu Installation Document
README.rst	Updating README.rst

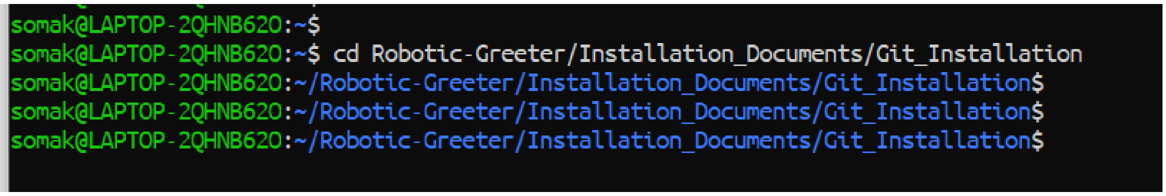
Use Git or checkout with SVN using the web URL.

https://github.com/ripanmukherjee/Roboti

3 days ago

Once the download is complete, please go to the following directory:

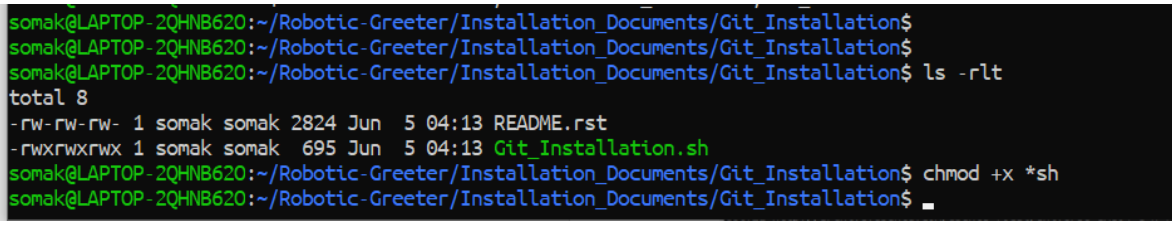
```
$ cd Robotic-Greeter/Installation_Documents/Git_Installation
```



```
somak@LAPTOP-2QHNB620:~$
somak@LAPTOP-2QHNB620:~$ cd Robotic-Greeter/Installation_Documents/Git_Installation
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
```

In Git_Installation folder you will get ***Git_Installation.sh*** script. To list the directories and files in this folder run “ls -lrt” and later change the executable permission for the file with “chmod”

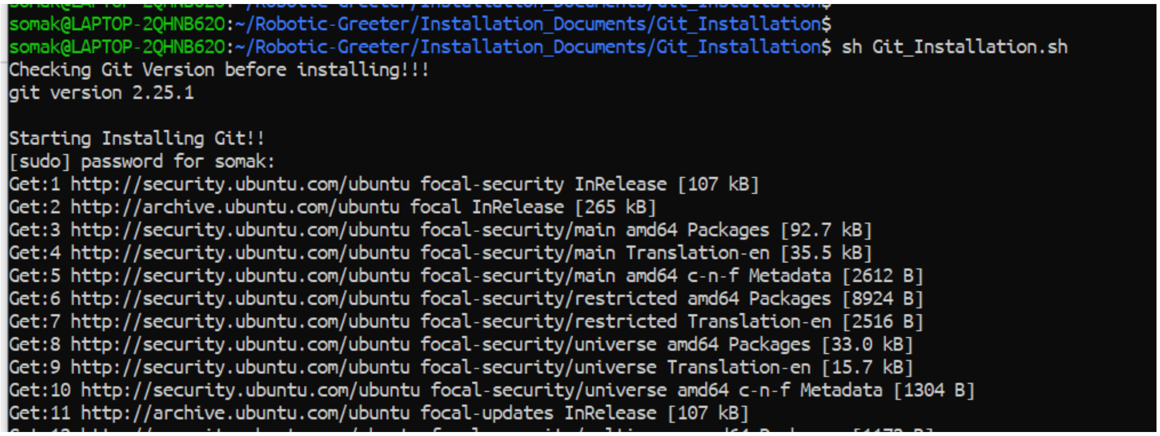
```
$ ls -lrt

$ chmod +x *.sh
```



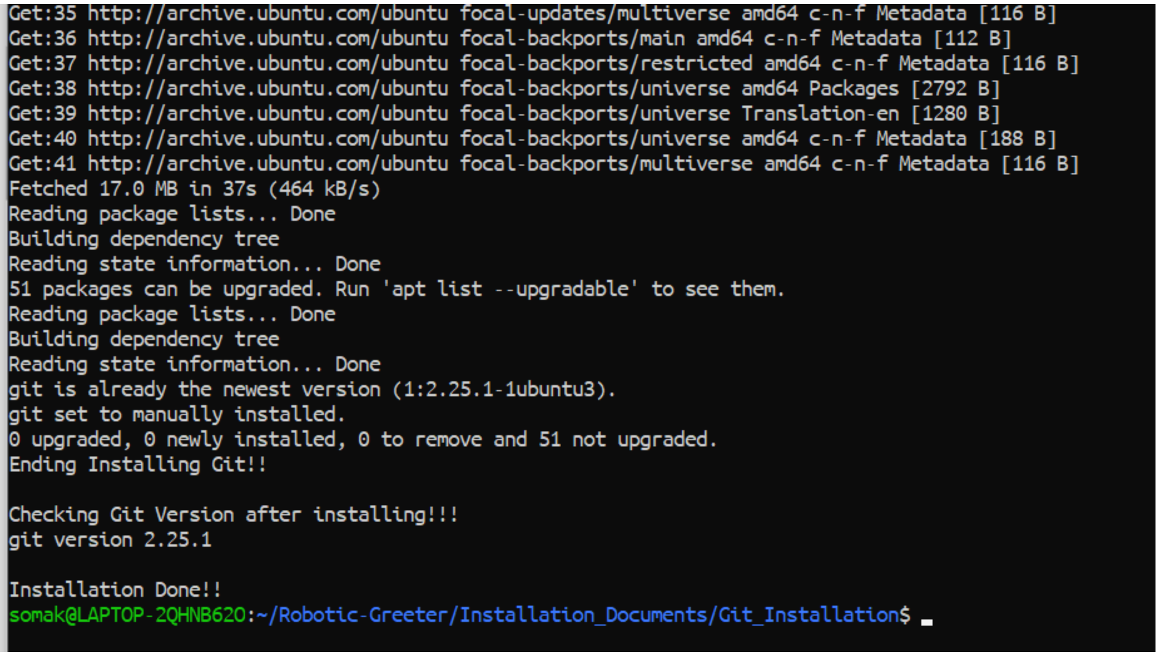
```
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ ls -rlt
total 8
-rw-rw-rw- 1 somak somak 2824 Jun  5 04:13 README.rst
-rwxrwxrwx 1 somak somak  695 Jun  5 04:13 Git_Installation.sh
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ chmod +x *sh
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
```

After that run, the scripts as follow:

```
$ sh Git_Installation.sh
```



```
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ sh Git_Installation.sh
Checking Git Version before installing!!!
git version 2.25.1

Starting Installing Git!!
[sudo] password for somak:
Get:1 http://security.ubuntu.com/ubuntu focal-security InRelease [107 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal InRelease [265 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [92.7 kB]
Get:4 http://security.ubuntu.com/ubuntu focal-security/main Translation-en [35.5 kB]
Get:5 http://security.ubuntu.com/ubuntu focal-security/main amd64 c-n-f Metadata [2612 B]
Get:6 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 Packages [8924 B]
Get:7 http://security.ubuntu.com/ubuntu focal-security/restricted Translation-en [2516 B]
Get:8 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [33.0 kB]
Get:9 http://security.ubuntu.com/ubuntu focal-security/universe Translation-en [15.7 kB]
Get:10 http://security.ubuntu.com/ubuntu focal-security/universe amd64 c-n-f Metadata [1304 B]
Get:11 http://archive.ubuntu.com/ubuntu focal-updates InRelease [107 kB]
```



```
Get:35 http://archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 c-n-f Metadata [116 B]
Get:36 http://archive.ubuntu.com/ubuntu focal-backports/main amd64 c-n-f Metadata [112 B]
Get:37 http://archive.ubuntu.com/ubuntu focal-backports/restricted amd64 c-n-f Metadata [116 B]
Get:38 http://archive.ubuntu.com/ubuntu focal-backports/universe amd64 Packages [2792 B]
Get:39 http://archive.ubuntu.com/ubuntu focal-backports/universe Translation-en [1280 B]
Get:40 http://archive.ubuntu.com/ubuntu focal-backports/universe amd64 c-n-f Metadata [188 B]
Get:41 http://archive.ubuntu.com/ubuntu focal-backports/multiverse amd64 c-n-f Metadata [116 B]
Fetched 17.0 MB in 37s (464 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
51 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree
Reading state information... Done
git is already the newest version (1:2.25.1-1ubuntu3).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 51 not upgraded.
Ending Installing Git!!

Checking Git Version after installing!!!
git version 2.25.1

Installation Done!!
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
```

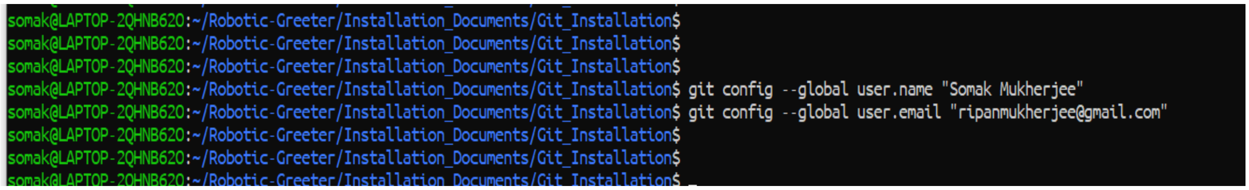
Installation done!!

After the installation done, you need to set some parameter (configuration) for the first time.

Working with Git & GitHub

The first thing you should do when you install Git is to set your username and email address. This is important because every Git commit uses this information, and it is immutably baked into the commits you start creating:

```
$ git config --global user.name "Example Name"

$ git config --global user.email "Example@email.com"
```

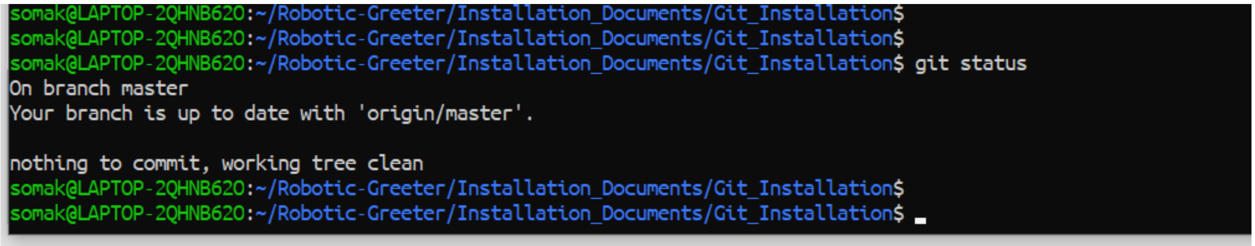


```
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ git config --global user.name "Somak Mukherjee"
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ git config --global user.email "ripanmukherjee@gmail.com"
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ _
```

Again, you need to do this only once if you pass the **--global** option, because then Git will always use that information for anything you do on that system. If you want to override this with a different name or email address for specific projects, you can run the command without the **--global** option when you are in that project.

Now, whenever you modify or add or remove any codes or directories or files then you need to check the status, add (add), rm (remove) and commit it as follow:

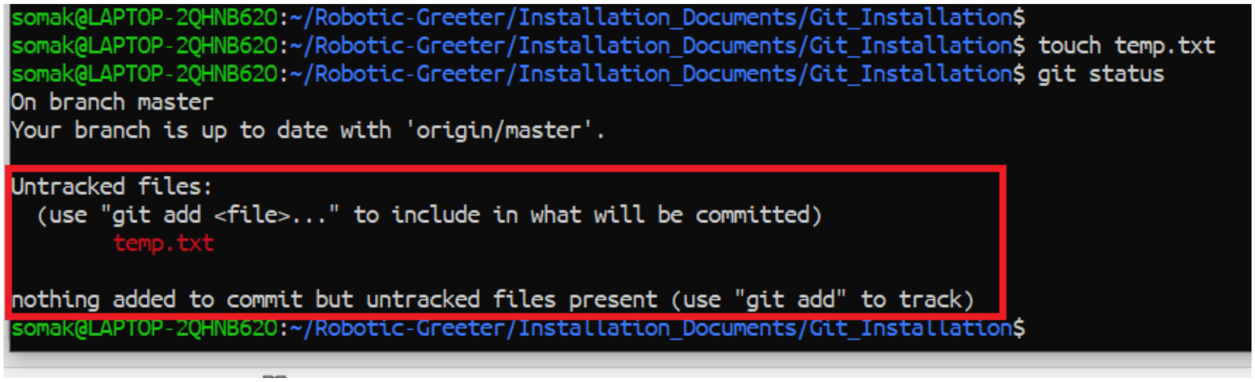
```
$ git status

$ git add example_code

$ git rm example_code

$ git status

$ git commit -a -m “Comment”
```



```
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ _
```

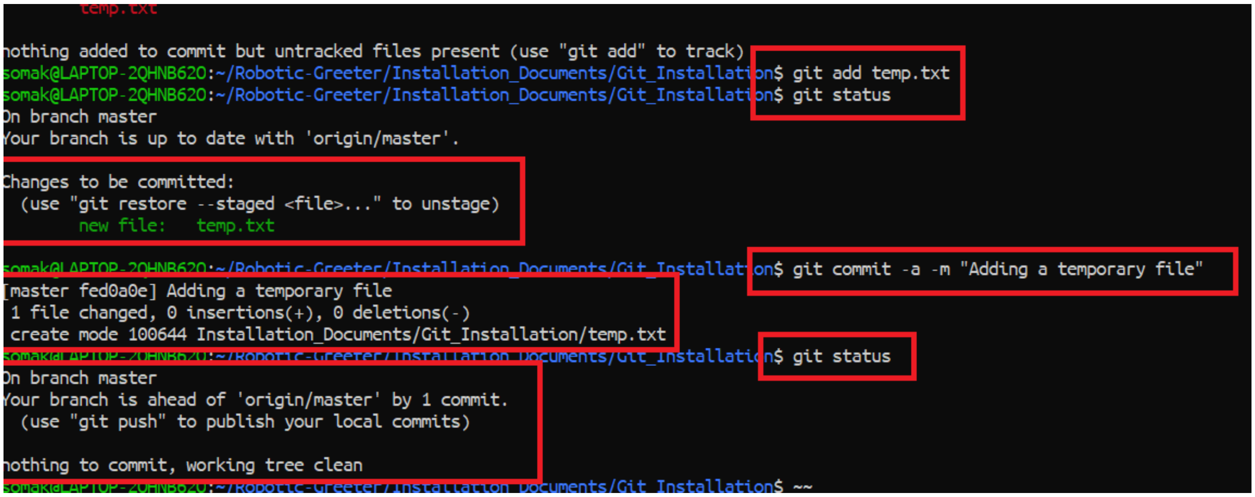
By running git status, you will get information that do you have anything to add or remove or commit it. If yes, then Git will show you that what you need to do if not then status will show as “nothing to commit”

As example if I add a temp.txt file and run the git status as follow:



```
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ touch temp.txt
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ git status
On branch master
Your branch is up to date with 'origin/master'.

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

nothing added to commit but untracked files present (use "git add" to track)
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$
```

Here you can see Git is showing Untracked file: temp.txt and it is showing (use "git add <file>..." to include in what will be committed). So, we can understand that we need to add the file and later commit it as below:



```
        temp.txt

nothing added to commit but untracked files present (use "git add" to track)
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ git add temp.txt
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   temp.txt

somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ git commit -a -m "Adding a temporary file"
[master fed0a0e] Adding a temporary file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Installation_Documents/Git_Installation/temp.txt
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
somak@LAPTOP-2QHNB620:~/Robotic-Greeter/Installation_Documents/Git_Installation$ ~~
```

You can do the same when you remove a files or directory. Also, Git give us option to **push** the new changes from your computer to GitHub repository (Cloud repository) and **pull** it from GitHub repository (Cloud repository) to your computer. But you can **push** and **pull** only to and from your GitHub repository. Since the other repository could be private or password protected. Because **push** and **pull** command will always ask for username and password of the GitHub page. You can do so as follow:

```
$ git push

$ git pull
```

For more details related to Git and GitHub please visit the following website:

https://git-scm.com/