Installing and using Git and GitHub on Ubuntu: A beginner's guide

GitHub is a treasure trove of some of the world's best projects, built by the contributions of developers all across the globe. This simple, yet extremely powerful platform helps every individual interested in building or developing something big to contribute and get recognized in the open source community.

On this page

This tutorial will be a quick setup guide for installing and using GitHub and how to perform its various functions of creating a repository locally, connecting this repo to the remote host that contains your project (where everyone can see), committing the changes and finally pushing all the content in the local system to GitHub.

Please note that this tutorial assumes that you have a basic knowledge of the terms used in Git such as push, pull requests, commit, repository, etc. It also requires you to register to GitHub here and make a note of your GitHub username. So let's begin:

1 Installing Git for Linux

Download and install Git for Linux :

```
sudo apt-get install git
```

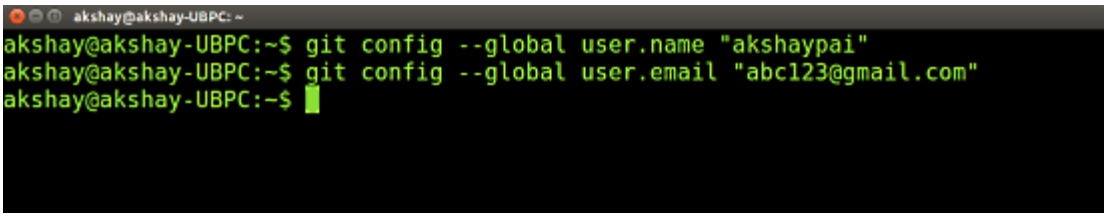
2 Configuring GitHub

Once the installation has successfully completed, the next thing to do is to set up the configuration details of the GitHub user. To do this use the following two commands by replacing "user_name" with your GitHub username and replacing "email_id" with your email-id you used to create your GitHub account.

```
git config --global user.name "user_name"
```

```
git config --global user.email "email_id"
```

The following image shows an example for my configuration with my "user_name" being "akshaypai" and my "email_id" being "abc123@gmail.com"



3 Creating a local repository

Create a folder in your system. This will serve as a local repository which will later be pushed onto the GitHub website. Use the following command:

```
git init Mytest
```

If the repository is created successfully, then you will get the following line:

```
Initialized empty Git repository in /home/akshay/Mytest/.git/
```

This line may vary depending on your system.

So here, Mytest is the folder that is created and "init" makes the folder a GitHub repository. Change the directory to this newly created folder:

```
cd Mytest
```

4 Creating a README file to describe the repository

Now create a README file and enter some text like "this is a git setup on linux". The README file is generally used to describe what the repository contains or what the project is all about. Example:

```
gedit README
```

You can use any other text editors. I use gedit. The content of the README file will be:

```
This is a git repo
```

5 Adding repository files to an index

This is an important step. Here we add all the things that need to be pushed onto the website into an index. These things might be the text files or programs that you might add for the first time into the repository or it could be adding a file that already exists but with some changes (a newer version/updated version).

Here we already have the README file. So, lets create another file which contains a simple C program and call it sample.c. The contents of it will be:

```
#include<stdio.h>
int main()
{
printf("hello world");
return 0;
}
```

So, now that we have 2 files

```
README and sample.c
```

add it to the index by using the following 2 commands:

```
git add README
```

```
git add smaple.c
```

Note that the "git add" command can be used to add any number of files and folders to the index. Here, when I say index, what I am refering to is a buffer like space that stores the files/folders that have to be added into the Git repository.

6 Committing changes made to the index

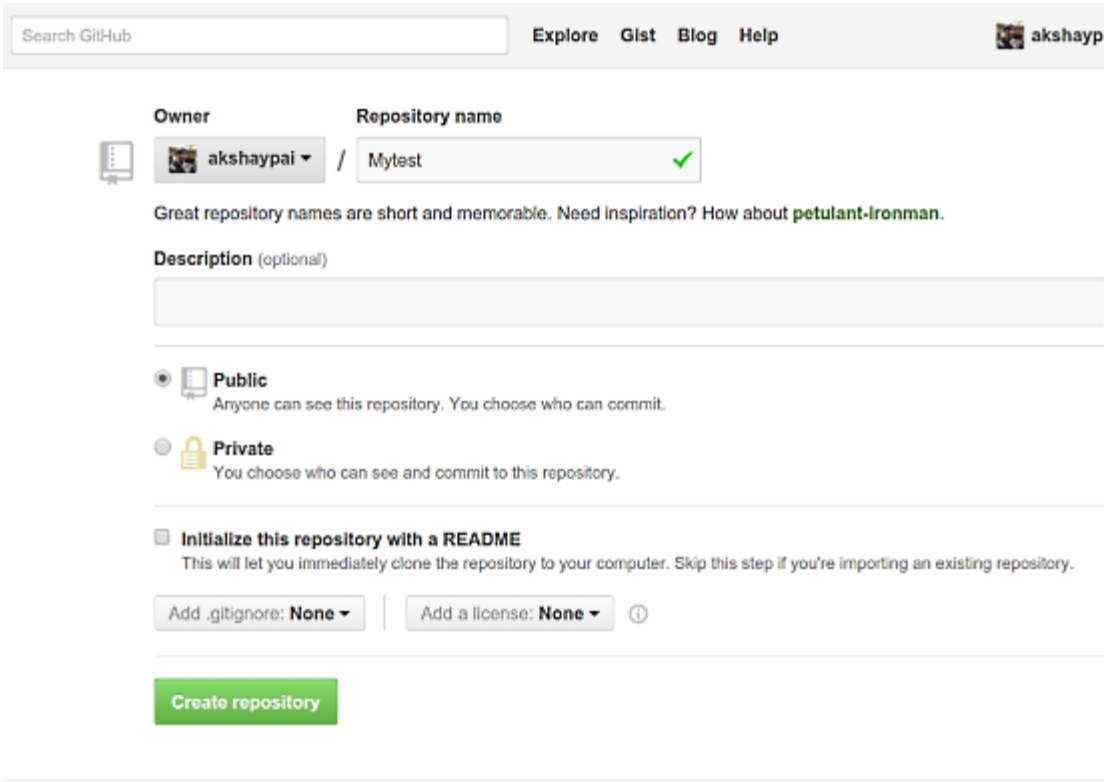
Once all the files are added, we can commit it. This means that we have finalized what additions and/or changes have to be made and they are now ready to be uploaded onto our repository. Use the command :

```
git commit -m "some_message"
```

"some_message" in the above command can be any simple message like "my first commit" or "edit in readme", etc.

7 Creating a repository on GitHub

Create a repository on GitHub. Notice that the name of the repository should be the same as the repository's on the local system. In this case, it will be "Mytest". To do this login to your account on https://github.com. Then click on the "plus(+)" symbol at the top right corner of the page and select "create new repository". Fill the details as shown in the image below and click on "create repository" button.



Once this is created, we can push the contents of the local repository onto the GitHub repository in your profile. Connect to the repository on GitHub using the command:

Important Note: Make sure you replace 'user_name' and 'Mytest' in the path with your Github username and folder before running the command!

```
git remote add origin https://github.com/user_name/Mytest.git
```

8 Pushing files in local repository to GitHub repository

The final step is to push the local repository contents into the remote host repository (GitHub), by using the command:

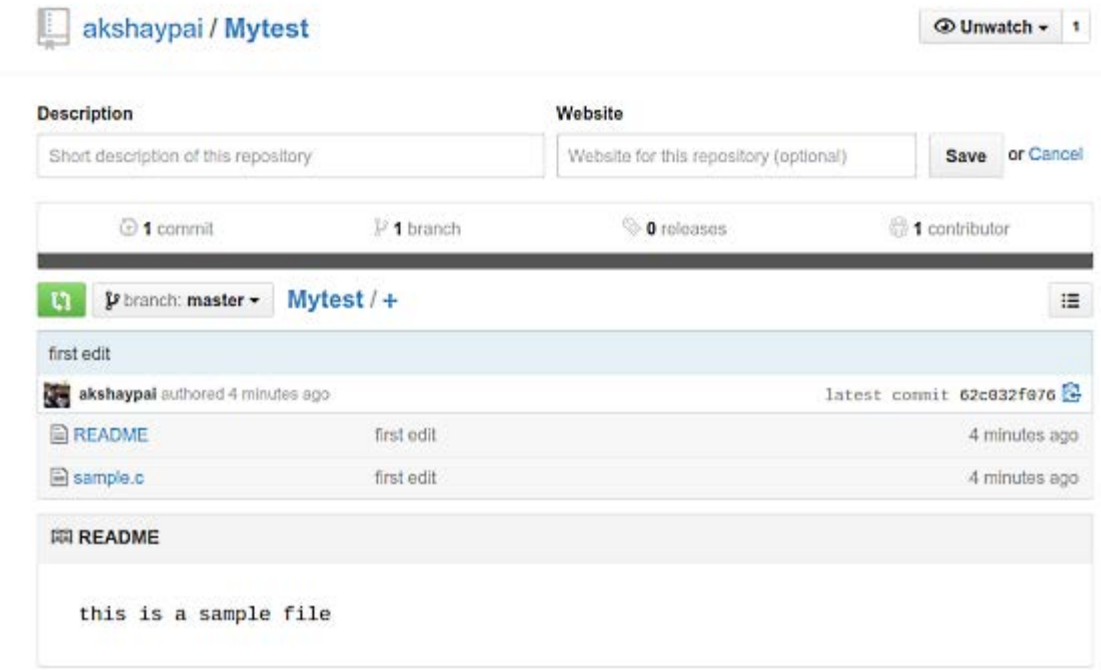
```
git push origin master
```

Enter the login credentials [user_name and password] .

The following image shows the procedure from step 5 to step 8

```
akshay@akshay-UBPC: ~/Mytest
akshay@akshay-UBPC:~/Mytest$ git add README
akshay@akshay-UBPC:~/Mytest$ git add sample.c
akshay@akshay-UBPC:~/Mytest$ git commit -m "first edit"
[master (root-commit) 62c032f] first edit
 2 files changed, 7 insertions(+)
 create mode 100644 README
 create mode 100644 sample.c
akshay@akshay-UBPC:~/Mytest$ git remote add origin https://github.com/akshaypai/
Mytest.git
akshay@akshay-UBPC:~/Mytest$ git push origin master
Username for 'https://github.com': akshaypai
Password for 'https://akshaypai@github.com':
Counting objects: 4, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 337 bytes | 0 bytes/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/akshaypai/Mytest.git
 * [new branch]      master -> master
akshay@akshay-UBPC:~/Mytest$
```

So this adds all the contents of the Mytest folder (my local repository) to GitHub. For subsequent projects or for creating repositories, you can start off with step 3 directly. Finally, if you login to to your GitHub account and click on your Mytest repository, you can see that the 2 files README and sample.c have been uploaded and are visible to all as shown in the follwoing image.



Links

- Git - Source Version Control System
- Github

view as pdf | print

Share this page: Tweet Follow

19

50 Comment(s)

Add comment

Name *

Email *

p

Submit comment

Comments

From: nwnpallewela Reply

This is great... Thanks for your help :)

From: Feragon Reply

Thanks a LOT!!! Very usefull tuto, and very simple. Thanks again.

From: Deepak Kanavikar Reply

This was precisely what i was looking for... Thanks

From: ashutoshh Reply

thanks a lot

From: honlulu Reply

<h1> very helpful xD Thanks a lot </h1>

From: karan Reply

awesome , thank you

From: Chris Reply

Thank you!

From: auchomage Reply

Thanks for this, it is very clear and helpful.

From: vartika Reply

thank you very much for such a clear and concise tutorial! :)

From: saurabh Reply

Thanks a lot, really very helpful.

From: Rajani Reply

Very helpful. Thank you very much

From: Kishor Reply

Best..

From: goldie Reply

awesome... spent so much time but did not get... but with this tutorial... its really easy.. thank u

From: jeet Reply

awesome easy to understand.....

From: swayne Reply

Very nice, concise beginners tutorial! FYI... one little typo found... In section 5: git add smaple.c

(should be "sample.c")

From: maitreyee Reply

Extremely helpful. I followed it step by step and I got exactly what I wanted

From: Jerry Reply

Bravo!!!

From: neil Reply

Awesome

From: Bilal Reply

Thank you so much

From: Bilal Reply

one more thing friends, suppose someone get any issue in git push origin master
please use this git push origin master --force its work for me

From: jorge8979787 Reply

thanks!

From: Luna Das Reply

nice article that what I was looking for :)

From: Murat Ersin Reply

It's work so great.Thanks for this tutorial.

From: om Reply

This is very cool tuto, love it.

From: Chichio Reply

Thanks. But i must run command: git pull origin master before git push origin masterI'm newbie

From: Chatchai Saratakij Reply

Wow, Thank you^_^

From: luan Reply

how to integrate postgresql to github?

From: Shradha Reply

Thank you so much..This is the most precisely explained tutorial...:)

From: Kautsya Kanu Reply

Best Tutorial that I found.. Thanks a lot!! :) You are really great..

From: yohannes Reply

Thanks that will help for starters like me.

From: Jacob Reply

This was great, but I initially received a error when I tried the git push origin master.
The error looks something like this: fatal: unable to access 'https://github.gatech.edu/jc89x0/SevFiewk.git/': server certificate verification failed. CAfile: /etc/ssl/certs/ca-certificates.crt CRLfile: none
I believe it is a certificate trust issue, but the fix for me was to enter the following commands (ref: http://stackoverflow.com/questions/21181231/server-certificate-verification-failed-cafile-etc-ssl-certs-ca-certificates-c)
export GIT_SSL_NO_VERIFY=1 #or git config --global http.sslverify falseAfterwards, I am presented with an option to provide my username and password.

From: Shashikant Singh Reply

Very good tutorial . Very clean explaination.

From: Devashish Kumar Jaiswal Reply

Thanks a lot..This is very helpful for everyone

From: Tingu Reply

Nice One Article

From: Tingu Reply

Masaalla article

From: Zakki Reply

Excellent tuto!

From: Graham Newman Reply

Brilliant - many thanks!

From: Ravi Kumar Reply

That's great...Thankyou sir

From: Ishadi Reply

Thanks a lot!

From: subhajyoti Reply

too good

From: Nisal Reply

This is great... Thanks for your help :)

From: Kevin Reply

That's amazing. The best hands on beginner's guide to git. It isn't much of a big deal. Thanks

From: Nikhil Chavda Reply

this tutorial is very usefull for me thank you so much.

From: ahahah Reply

This was soooo helpfull. Thank you soooo much. :))

From: Victor Reply

Thanks for this , I never thought it was so simple, you helpme so much

From: WRONG! Reply

remote add orgin, not add remote origin

From: Irishologram Reply

I just followed this tutorial and create a new github repository, thank you!

From: KC Reply

Thank You! This has been of real help.

From: murali Reply

Thanks for the usefull information....

From: Shrikrishna Reply

Thanks mate exellent article.

Tutorial Info

Author: Akshay Pai
Tags: ubuntu, linux, programming

Share This Page