Name: Saklen Mujawar
Roll No: 612037

Experiment no. 02

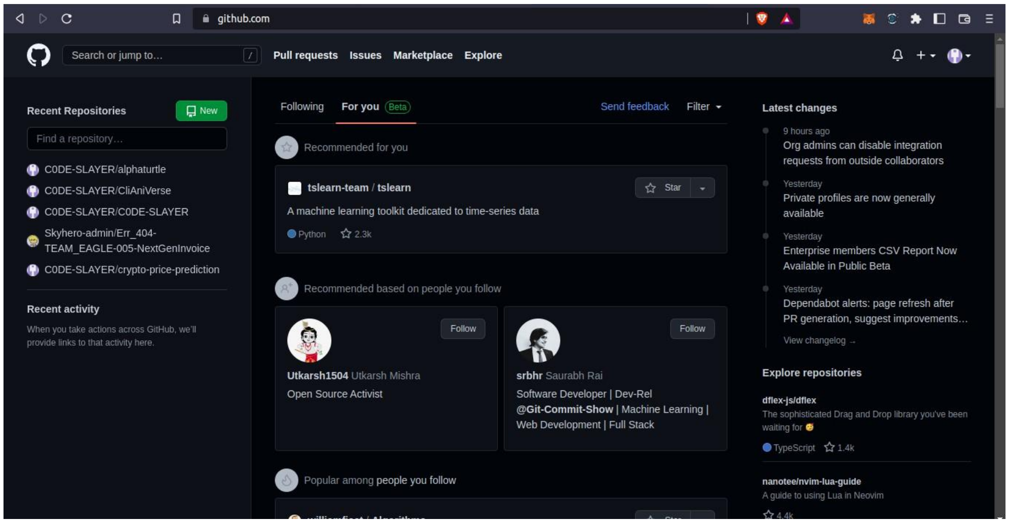
Aim: To perform version control on a website or software using the git version control tool.

Theory:

- Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later. For the examples in this book, you will use software source code as the files being version controlled, though in reality you can do this with nearly any type of file on a computer.
- A component of software configuration management, version control, also known as revision control or source control, is the management of changes to documents, computer programs, large web sites, and other collections of information.
- Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.Git is easy to learn and has a tiny footprint with lightning fast performance. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like cheap local branching, convenient staging areas, and multiple workflows.
- Founded in April 2008, GitHub is a web-based hosting service where anyone can share programming code with anyone else. GitHub offers their services for free to the general public and for businesses, they offer paid service plans. GitHub also offers a service called GitHub Gist, which is a Pastebin-like service to paste and quickly share snippets of your code.GitHub was started in 2008 and is based on a code management system developed by Linus Torvalds, called Git. Utilizing GitHub's hosting service provides users with revision control for their code, allowing them and others to view all revisions of the code shared on the site.

Steps to install and implement version control on Github using git:

a. Login to your Github account or if you don't have one go ahead and create one. After login click on the green button which say New



Recent Repositories

Find a repository…

CODE-SLAYER/alphaturtle
CODE-SLAYER/CliAniVerse
CODE-SLAYER/CODE-SLAYER
Skyhero-admin/Err_404-TEAM_EAGLE-005-NextGenInvoice
CODE-SLAYER/crypto-price-prediction

Recent activity

When you take actions across GitHub, we'll provide links to that activity here.

Recommended for you

tslearn-team / tslearn
A machine learning toolkit dedicated to time-series data
Python 2.3k

Recommended based on people you follow

Utkarsh1504 Utkarsh Mishra
Open Source Activist

srbhr Saurabh Rai
Software Developer | Dev-Rel @Git-Commit-Show | Machine Learning | Web Development | Full Stack

Popular among people you follow

Latest changes

9 hours ago
Org admins can disable integration requests from outside collaborators

Yesterday
Private profiles are now generally available

Yesterday
Enterprise members CSV Report Now Available in Public Beta

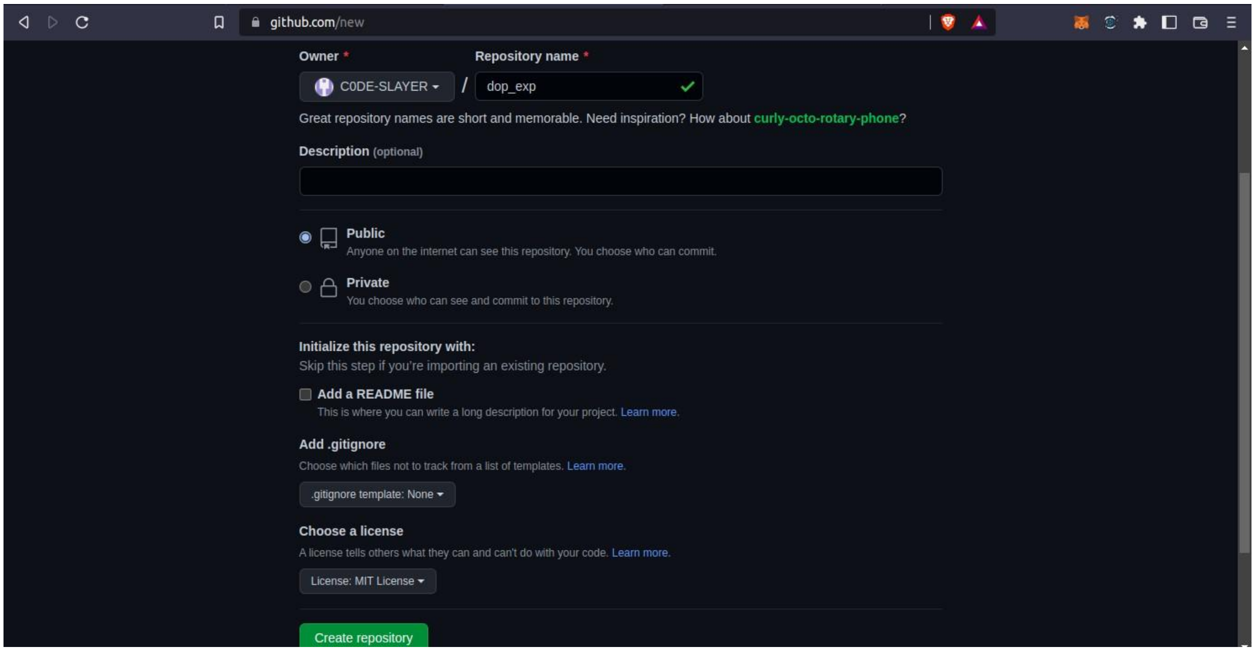
Yesterday
Dependabot alerts: page refresh after PR generation, suggest improvements…

Explore repositories

dflex-js/dflex
The sophisticated Drag and Drop library you've been waiting for 🥳
TypeScript 1.4k

nanotee/nvim-lua-guide
A guide to using Lua in Neovim
4.4k

b. Now select a name for your repo and leave the rest as default. Click on Create repository to create a repo



Owner * CODE-SLAYER / Repository name * dop_exp ✓

Great repository names are short and memorable. Need inspiration? How about curly-octo-rotary-phone?

Description (optional)

● Public
Anyone on the internet can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

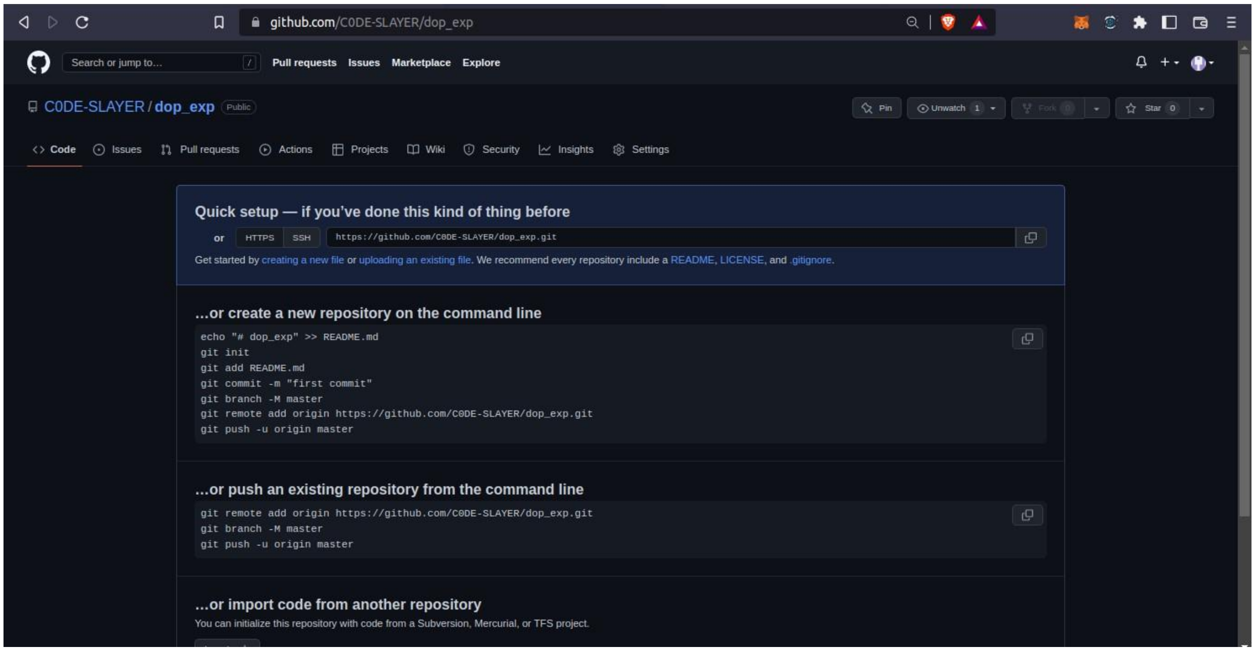
Initialize this repository with:
Skip this step if you're importing an existing repository.

□ Add a README file
This is where you can write a long description for your project. Learn more.

Add .gitignore
Choose which files not to track from a list of templates. Learn more.
.gitignore template: None

Choose a license
A license tells others what they can and can't do with your code. Learn more.
License: MIT License

c. We have successfully created our first repo on Github



Quick setup — if you've done this kind of thing before

https://github.com/C0DE-SLAYER/dop_exp.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

…or create a new repository on the command line

```
echo "# dop_exp" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M master
git remote add origin https://github.com/C0DE-SLAYER/dop_exp.git
git push -u origin master
```

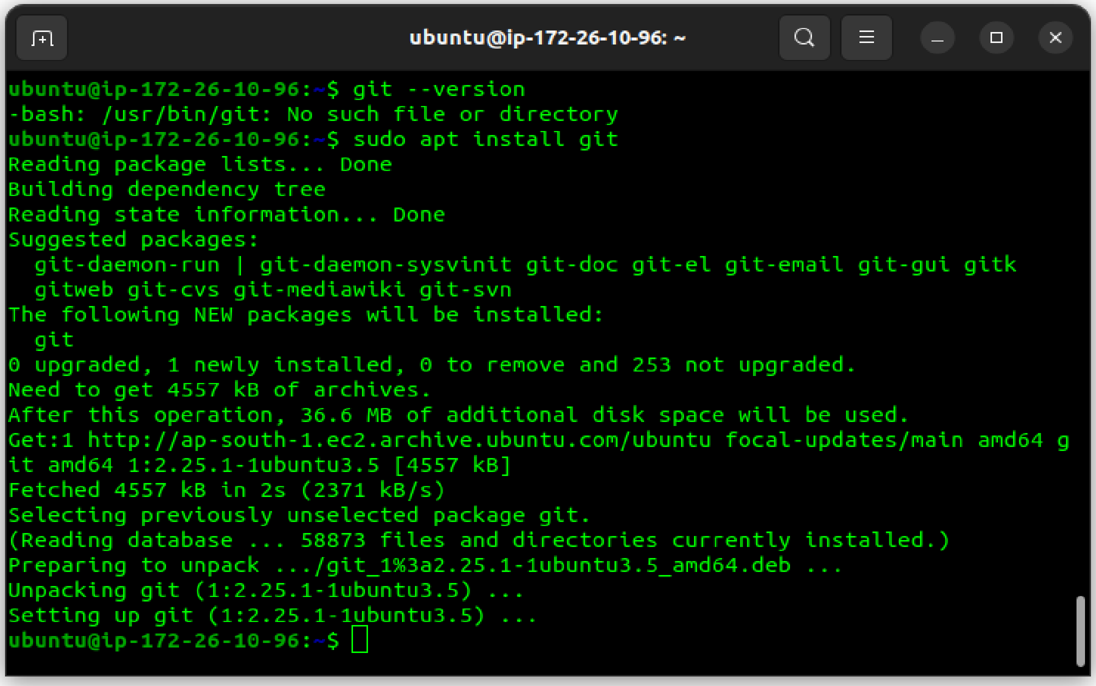
…or push an existing repository from the command line

```
git remote add origin https://github.com/C0DE-SLAYER/dop_exp.git
git branch -M master
git push -u origin master
```

…or import code from another repository

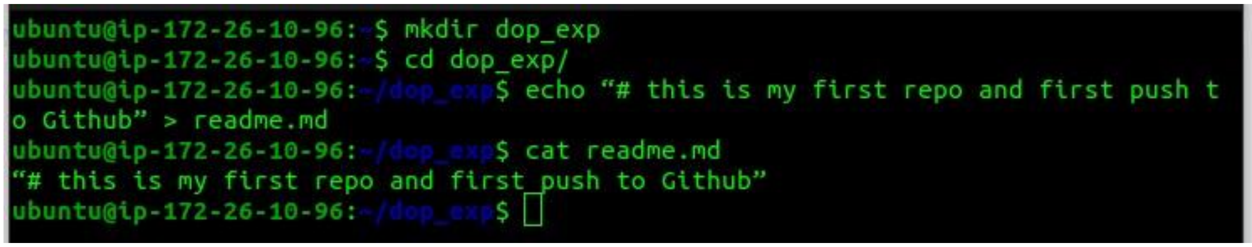
You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

d. Now open your terminal and check if you have git install or not using "git –version" and if git is not install use "sudo apt install git"



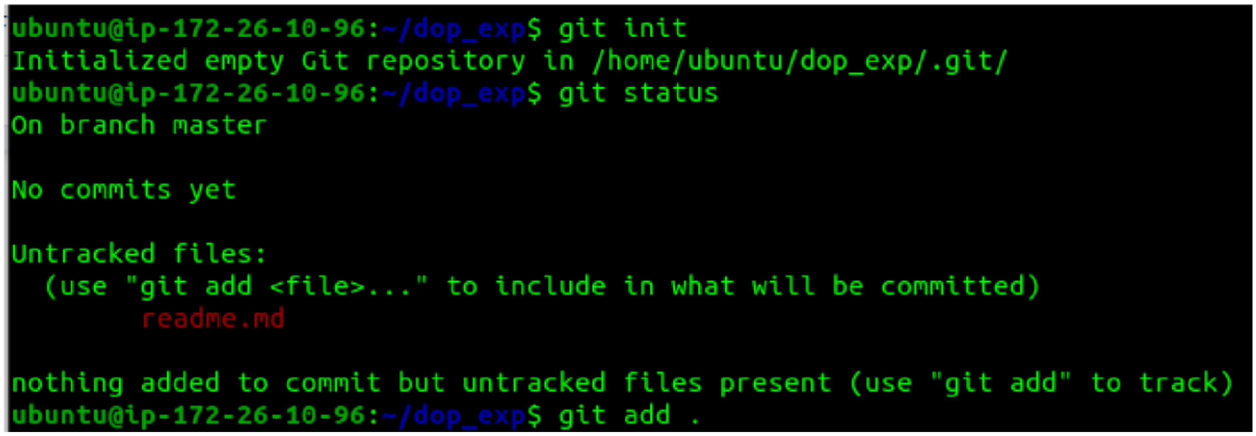
```
ubuntu@ip-172-26-10-96:~$ git --version
-bash: /usr/bin/git: No such file or directory
ubuntu@ip-172-26-10-96:~$ sudo apt install git
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk
  gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git
0 upgraded, 1 newly installed, 0 to remove and 253 not upgraded.
Need to get 4557 kB of archives.
After this operation, 36.6 MB of additional disk space will be used.
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 g
it amd64 1:2.25.1-1ubuntu3.5 [4557 kB]
Fetched 4557 kB in 2s (2371 kB/s)
Selecting previously unselected package git.
(Reading database ... 58873 files and directories currently installed.)
Preparing to unpack .../git_1%3a2.25.1-1ubuntu3.5_amd64.deb ...
Unpacking git (1:2.25.1-1ubuntu3.5) ...
Setting up git (1:2.25.1-1ubuntu3.5) ...
ubuntu@ip-172-26-10-96:~$ 
```

e. Create a new dir using “mkdir dop_exp” and write the following command to create and write content on file ‘echo “# this is my first repo and first push to Github” > readme.md’



```
ubuntu@ip-172-26-10-96:~$ mkdir dop_exp
ubuntu@ip-172-26-10-96:~$ cd dop_exp/
ubuntu@ip-172-26-10-96:~/dop_exp$ echo “# this is my first repo and first push t
o Github” > readme.md
ubuntu@ip-172-26-10-96:~/dop_exp$ cat readme.md
“# this is my first repo and first push to Github”
ubuntu@ip-172-26-10-96:~/dop_exp$ 
```

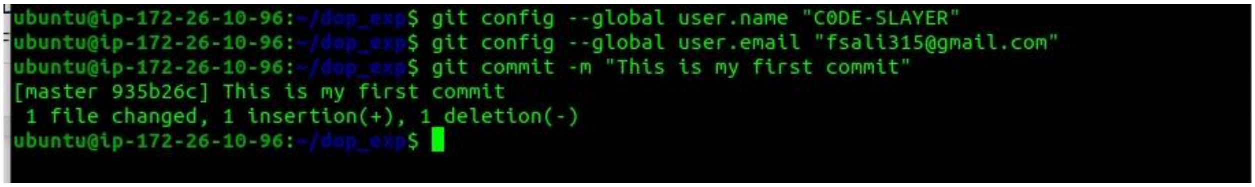
f. To make a dir a git repo use “git init”. To check the status of dir use “git status”. To track files use “git add .” this will track all the files in that dir.



```
ubuntu@ip-172-26-10-96:~/dop_exp$ git init
Initialized empty Git repository in /home/ubuntu/dop_exp/.git/
ubuntu@ip-172-26-10-96:~/dop_exp$ git status
On branch master

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)
ubuntu@ip-172-26-10-96:~/dop_exp$ git add .
```

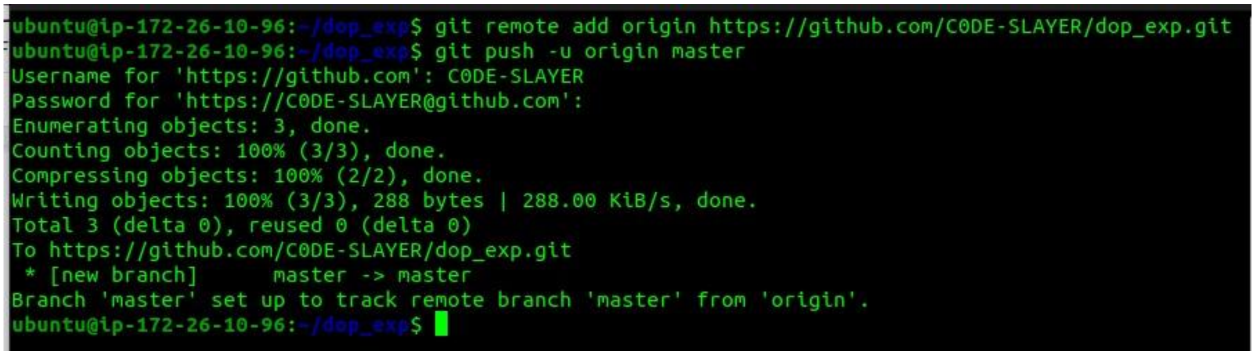
g. Set your user name using “git config --global user.name "C0DE-SLAYER"’.

Set your email using ‘git config --global user.email "fsali315@gmail.com"’ setting username and email will help to see who push the code And at last commit you change using ‘git commit -m “This is my first commit”’



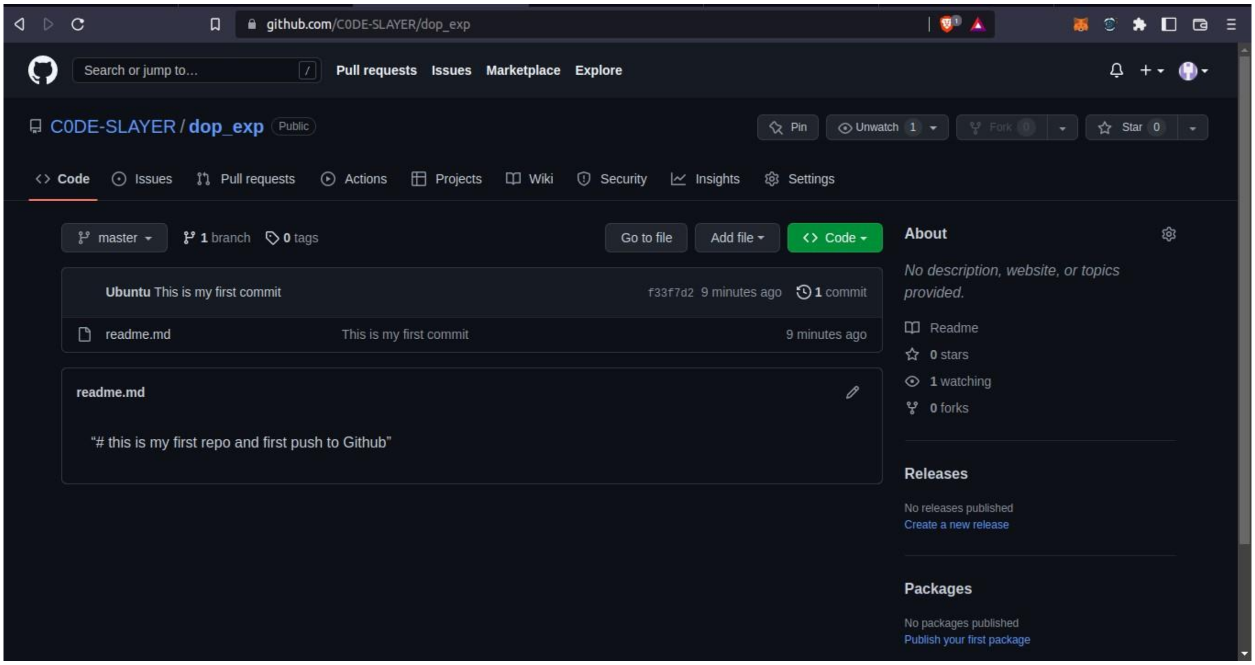
```
ubuntu@ip-172-26-10-96:~/dop_exp$ git config --global user.name "C0DE-SLAYER"
ubuntu@ip-172-26-10-96:~/dop_exp$ git config --global user.email "fsali315@gmail.com"
ubuntu@ip-172-26-10-96:~/dop_exp$ git commit -m "This is my first commit"
[master 935b26c] This is my first commit
 1 file changed, 1 insertion(+), 1 deletion(-)
ubuntu@ip-172-26-10-96:~/dop_exp$ 
```

h. Using this command you will connect your local repo to you github repo “git remote add origin https://github.com/C0DE-SLAYER/dop_exp.git” and using this command you can push you committed code to github repo “git push -u origin master” it will ask your username and password when you provide it your code will be push to your github account.



```
ubuntu@ip-172-26-10-96:~/dop_exp$ git remote add origin https://github.com/C0DE-SLAYER/dop_exp.git
ubuntu@ip-172-26-10-96:~/dop_exp$ git push -u origin master
Username for 'https://github.com': C0DE-SLAYER
Password for 'https://C0DE-SLAYER@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 288 bytes | 288.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/C0DE-SLAYER/dop_exp.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
ubuntu@ip-172-26-10-96:~/dop_exp$
```

i. Now go to your github repo and see the readme file will be upload there



Ubuntu This is my first commit f33f7d2 9 minutes ago 1 commit

readme.md This is my first commit 9 minutes ago

readme.md

"# this is my first repo and first push to Github"

No description, website, or topics provided.

Readme

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Conclusion: We performed version control on website using git version control tool