

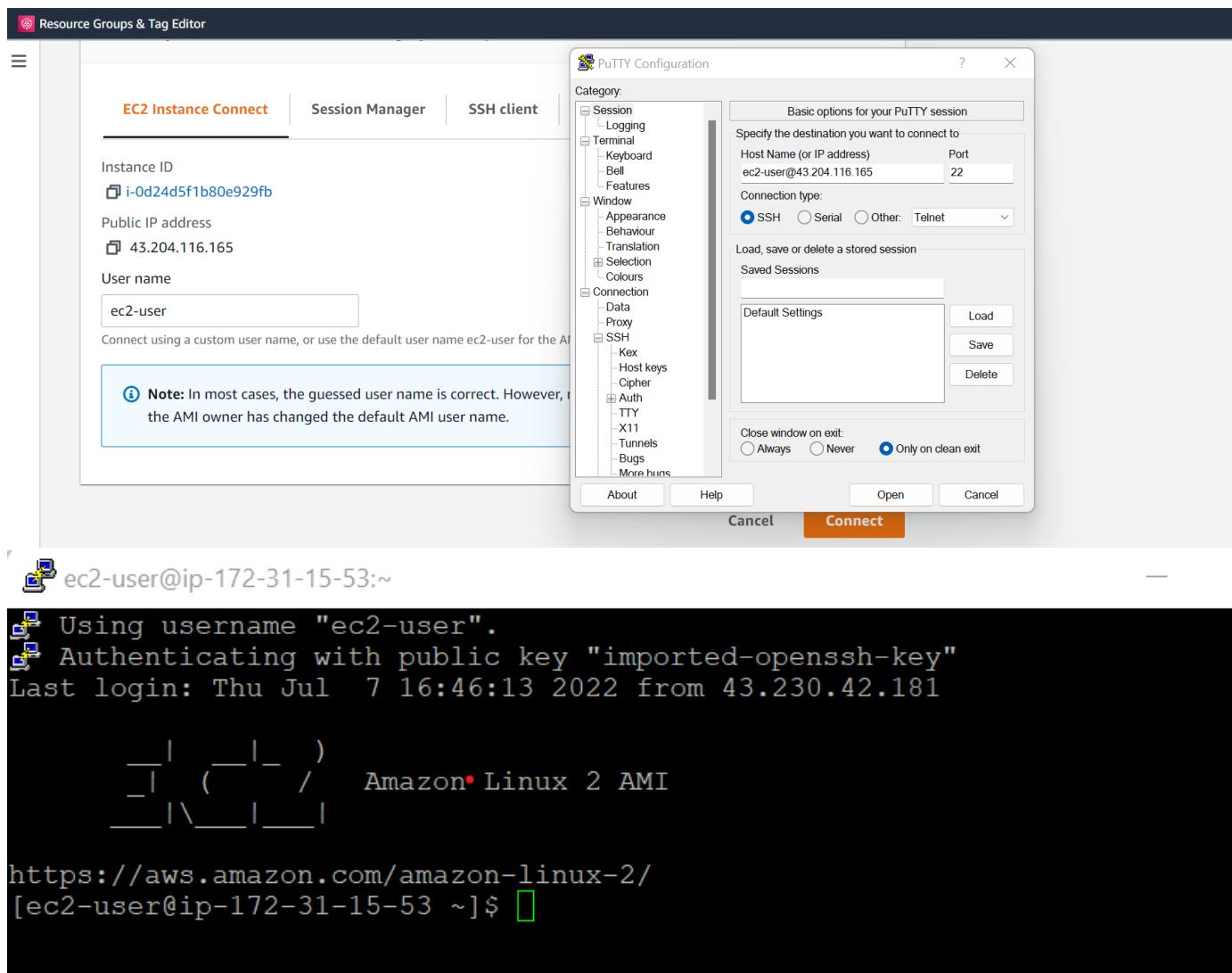
GIT AND GITHUB LAB

We have to launch two ec2-instances one will be in Mumbai region other will be N.Virginia region. Just in security group we have to add Http port 80 also, else other steps will be same.

Once Both Instances are ready then first connect to Mumbai region ec2-instance and run following command one after other.

These command will help you to setup git in Mumbai ec2-instance. Once we will done with setup in Mumbai Ec2-instance then we will again run same command to N.virginia region Ec2-instance also to configure git there.

```
sudo su root
yum update -y
yum install git -y
which git
git --version
git config --global user.name "mumbaiemp"
git config --global user.email "mumbaiemp@gmail.com"
git config --list
```



The screenshot shows the AWS Resource Groups & Tag Editor interface. On the left, there's a sidebar with three tabs: 'EC2 Instance Connect' (which is selected and highlighted in orange), 'Session Manager', and 'SSH client'. The main area displays an EC2 instance with the following details:

- Instance ID: i-0d24d5f1b80e929fb
- Public IP address: 43.204.116.165
- User name: ec2-user

A note in the interface states: "Note: In most cases, the guessed user name is correct. However, if the AMI owner has changed the default AMI user name."

A 'Putty Configuration' window is overlaid on the interface. The 'Session' category is selected. The 'Host Name (or IP address)' field is set to 'ec2-user@43.204.116.165' and the 'Port' is set to '22'. The 'Connection type' is set to 'SSH'. The 'Saved Sessions' section shows 'Default Settings'. At the bottom, the 'Close window on exit' options are set to 'Only on clean exit'.

The terminal window shows the following output:

```
ec2-user@ip-172-31-15-53:~$ Using username "ec2-user".
ec2-user@ip-172-31-15-53:~$ Authenticating with public key "imported-openssh-key"
Last login: Thu Jul  7 16:46:13 2022 from 43.230.42.181

[ec2-user@ip-172-31-15-53 ~]$
```

```
Using username "ec2-user".
Authenticating with public key "imported-ssh-key"
Last login: Thu Jul  7 16:46:13 2022 from 43.230.42.181

      _\|_(_\|_ /  Amazon Linux 2 AMI
      \_\|_\|_\|
```

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-15-53 ~]$ sudo su root
[root@ip-172-31-15-53 ec2-user]# 
```

```
Using username "ec2-user".
Authenticating with public key "imported-ssh-key"
Last login: Thu Jul  7 16:46:13 2022 from 43.230.42.181
```

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-15-53 ~]$ sudo su root
[root@ip-172-31-15-53 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[root@ip-172-31-15-53 ec2-user]# 
```

GIT AND GITHUB LAB

We have to launch two ec2-instances on

security group we have to add Http po

Once Both Instances are ready then fir

command one after other.

These command will help you to setup

Ec2-instance then we will again run sam

```
Using username "ec2-user".
Authenticating with public key "imported-ssh-key"
Last login: Thu Jul  7 16:46:13 2022 from 43.230.42.181
```

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-15-53 ~]$ sudo su root
[root@ip-172-31-15-53 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[root@ip-172-31-15-53 ec2-user]# yum install git -y
```

```
1 sudo su root
2 yum update -y
3 yum install git -y
4 git --version
5 git config --global username "mumbaiemp"
6 git config --global useremail "mumbaiemp@gmail.com"
```

```
[root@ip-172-31-15-53 ec2-user]# which git
/bin/git
[root@ip-172-31-15-53 ec2-user]# git --version
git version 2.32.0
[root@ip-172-31-15-53 ec2-user]# git config --global user.name "mumbaiemp"
[root@ip-172-31-15-53 ec2-user]# git config --global user.email "mumbaiemp@gmail.com"
[root@ip-172-31-15-53 ec2-user]# git config --list
user.name=mumbaiemp
user.email=mumbaiemp@gmail.com
[root@ip-172-31-15-53 ec2-user]# 
```

Now switch to N.virginia region connect to that ec2-instance and run all the command their also.

The screenshot shows the AWS Cloud9 IDE interface. At the top, there is a search bar and a navigation bar with 'N. Virginia' and 'kroydevops'. Below the search bar, there is a 'Instances (1/1)' section showing one instance named 'i-05910337bd317ff6c' which is 'Running' and of type 't2.micro'. The status is 'Initializing' with 'No alarms'. A 'Launch instances' button is also present. Below this, there is a 'Services' tab and a 'Resource Groups & Tag Editor' tab. A 'PuTTY Configuration' window is open in the foreground, showing the 'Session' category. The 'Host Name (or IP address)' is set to 'ec2-user@54.226.31.147' and the 'Port' is '22'. The 'Connection type' is set to 'SSH'. The 'Saved Sessions' section shows a single entry 'Default Settings'. The 'Close window on exit' option is set to 'Only on clean exit'. At the bottom of the PuTTY window are 'About', 'Help', 'Open', 'Cancel', and a large 'Connect' button.

```
ec2-user@ip-172-31-28-241:~$ 
[ec2-user@ip-172-31-28-241:~]$ Using username "ec2-user".
[ec2-user@ip-172-31-28-241:~]$ Authenticating with public key "imported-openssh-key"
Last login: Fri Jul  8 03:05:10 2022 from 203.189.248.184

  _\|_(_\|_ /     Amazon Linux 2 AMI
  _\|_\|_\|_|
[ec2-user@ip-172-31-28-241 ~]$ 
```

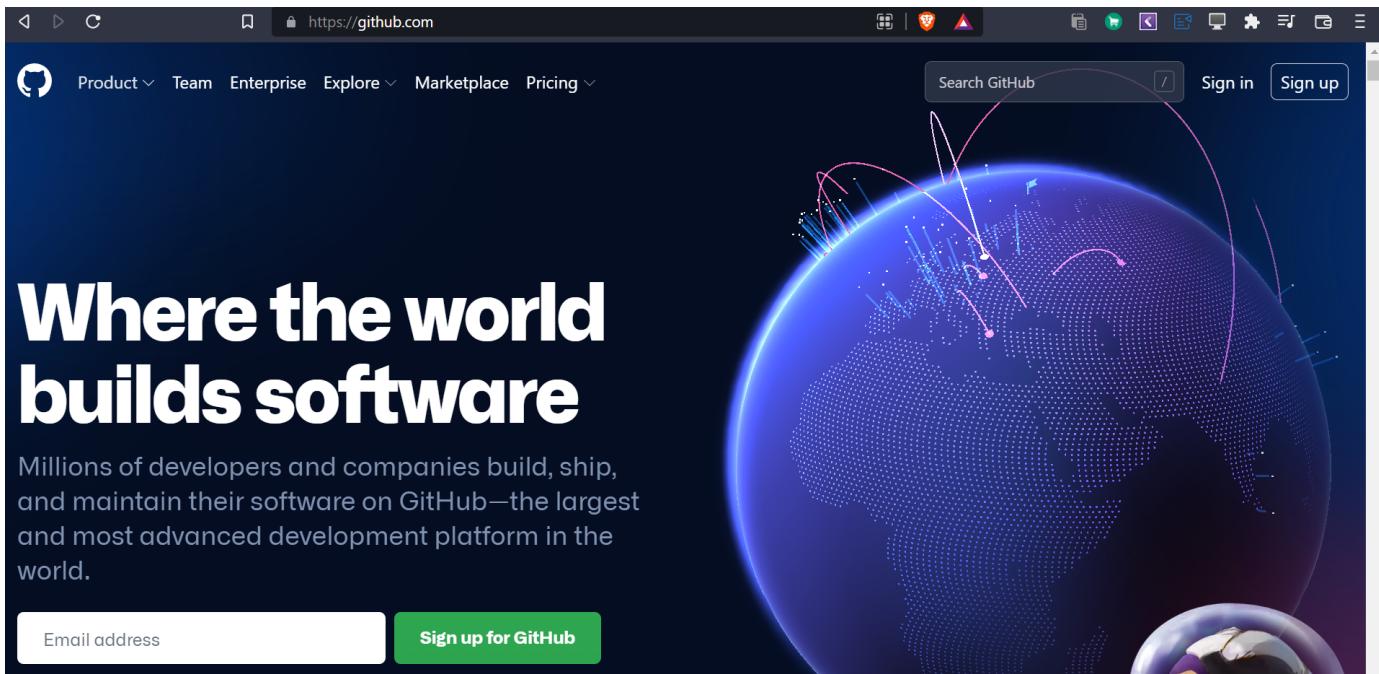
```
sudo su root
yum update -y
yum install git -y
which git
git --version
git config --global user.name "usemp"
git config --global user.email "usemp@gmail.com"
git config --list
```

```
Complete!
[root@ip-172-31-28-241 ec2-user]# which git
/bin/git
[root@ip-172-31-28-241 ec2-user]# git --version
git version 2.32.0
[root@ip-172-31-28-241 ec2-user]# git config --global user.name "usemp"
[root@ip-172-31-28-241 ec2-user]# git config --global user.email "usemp@gmail.co
m"
[root@ip-172-31-28-241 ec2-user]# git config --list
user.name=usemp
user.email=usemp@gmail.com
[root@ip-172-31-28-241 ec2-user]# 
```

Now we have successfully installed git and configured in both the Ec2-instance.

Now after setting up git, we will require a github account (central repository).

1. Go to link <https://github.com/>



Where the world builds software

Millions of developers and companies build, ship, and maintain their software on GitHub—the largest and most advanced development platform in the world.

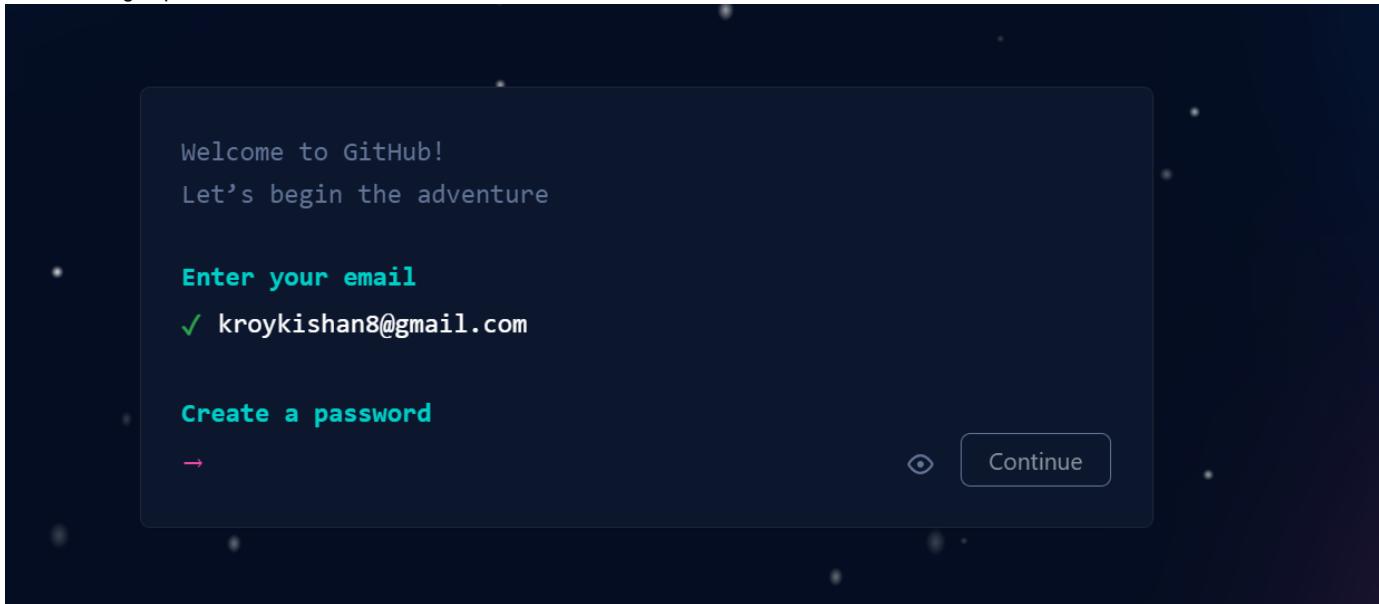
Email address [Sign up for GitHub](#)

Search GitHub

Sign in Sign up

The screenshot shows the GitHub homepage. At the top, there is a navigation bar with links for Product, Team, Enterprise, Explore, Marketplace, and Pricing. On the right side of the header is a search bar labeled "Search GitHub" with a magnifying glass icon, and buttons for "Sign in" and "Sign up". The main visual is a large, glowing blue globe representing the world, with a network of lines and dots indicating global connectivity. Below the globe, there is a call-to-action section with an input field for "Email address" and a green "Sign up for GitHub" button. The text "Where the world builds software" is prominently displayed in large white letters. A subtext below it reads: "Millions of developers and companies build, ship, and maintain their software on GitHub—the largest and most advanced development platform in the world." The URL "https://github.com" is visible in the browser's address bar.

2. click on Sign up



Welcome to GitHub!
Let's begin the adventure

Enter your email
✓ kroykishan8@gmail.com

Create a password

→ Continue

The screenshot shows the GitHub sign-up form. It starts with a welcome message: "Welcome to GitHub! Let's begin the adventure". Below this, there is a field for "Enter your email" with the value "✓ kroykishan8@gmail.com" followed by a checkmark. There is also a field for "Create a password" with a right-pointing arrow icon. At the bottom right of the form is a "Continue" button. The background is dark, and the overall design is clean and modern.

Welcome to GitHub!
Let's begin the adventure

Enter your email
✓ kroykishan8@gmail.com

Create a password
✓ ······

Enter a username
→ kroykishan8

Continue

Enter your email
✓ kroykishan8@gmail.com

Create a password
✓ ······

Enter a username
✓ kroykishan8

Would you like to receive product updates and announcements via email?

Type "y" for yes or "n" for no

→ n

Continue

Verify your account

Please solve this puzzle to verify that you are human

Click "Start puzzle" to continue

Start puzzle



Create account

You're almost done!
We sent a launch code to kroykishan8@gmail.com

→ [Enter code](#)



Didn't get your email? [Resend the code](#) or update your email address.



Welcome to GitHub

We are glad you're here.

How many team members will be working with you?

This will help us guide you to the tools that are best suited for your projects.

Just me

2 - 5

5 - 10

10 - 20

20 - 50

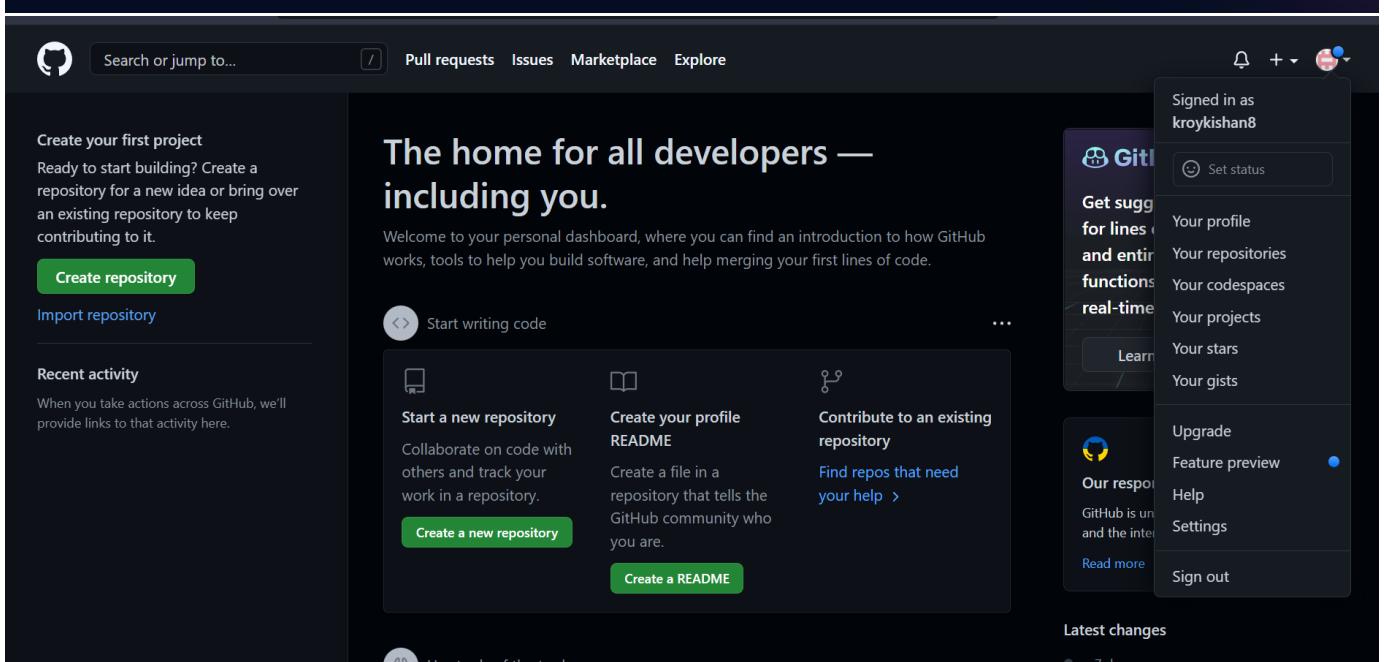
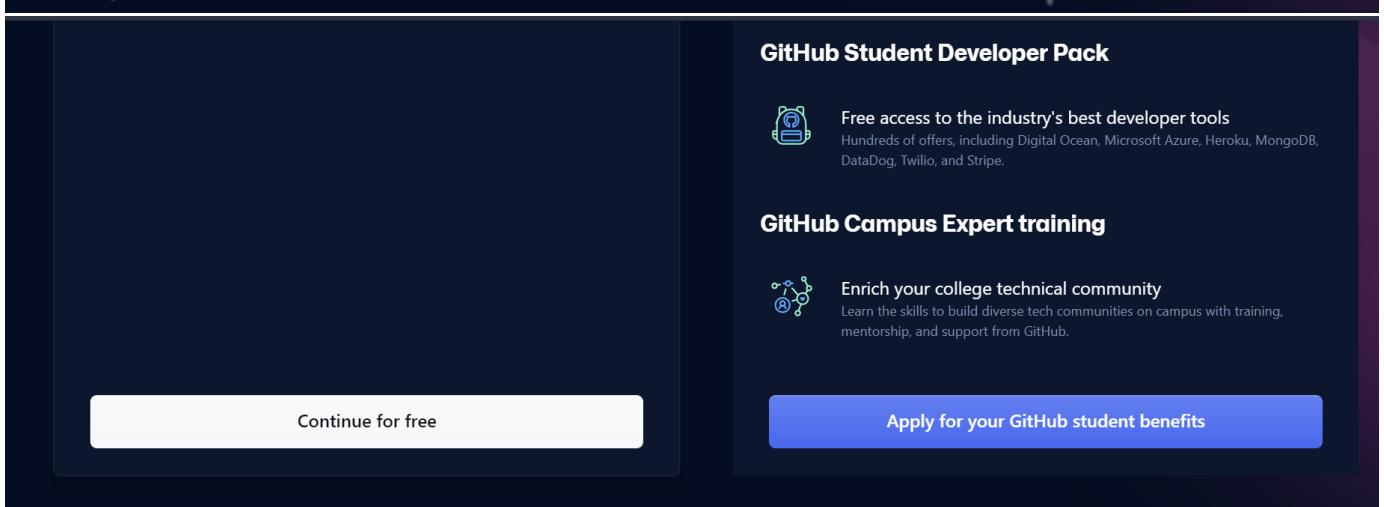
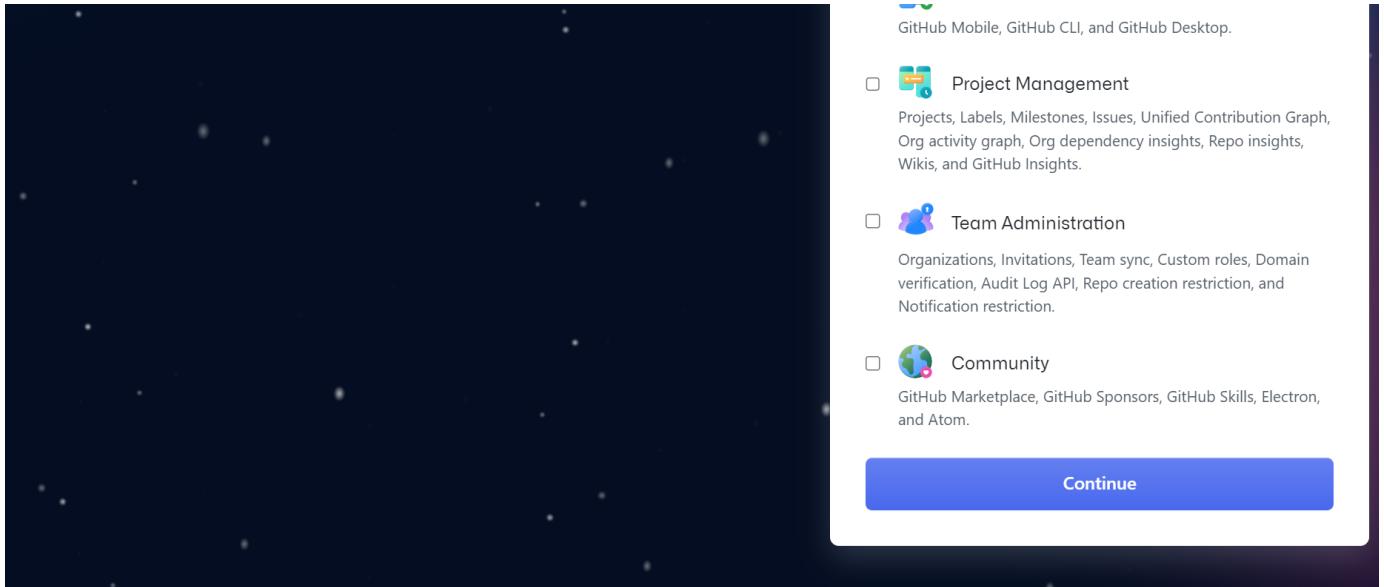
50+

Are you a student or teacher?

Student

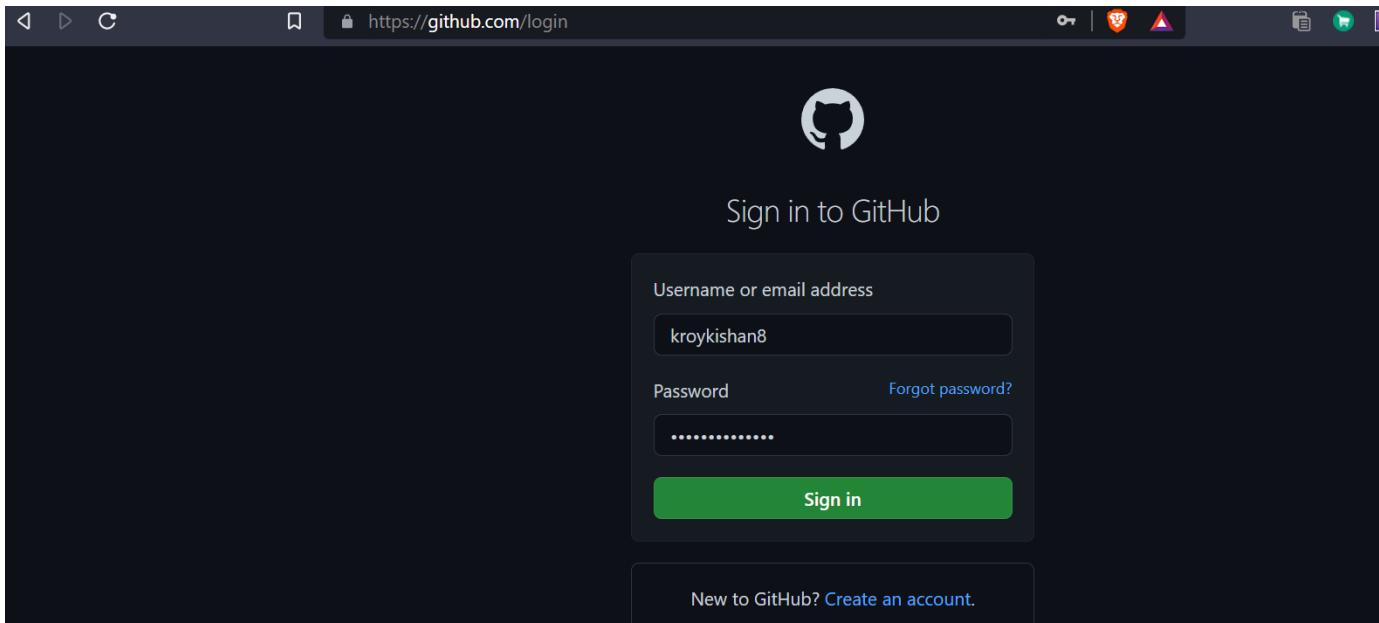
Teacher

[Continue](#)



Now you GitHub account is also setup. You can treat this github account as company account.

You can login to your github account anytime <https://github.com/login>



A screenshot of the GitHub dashboard. At the top, there is a search bar, a notifications icon, and a "+" button. The main content area features a large banner with the text "The home for all developers — including you." and a subtext about the personal dashboard. Below the banner are several cards: "Create your first project" (with a "Create repository" button), "Import repository", "Recent activity" (with a note about recent actions), "Start writing code" (with a "Start writing code" button), "Start a new repository" (with a "Create a new repository" button), "Create your profile" (with a "Create a README" button), "Contribute to an existing repository", and "Find repos that need your help". On the right side, there are two floating cards: "GitHub Copilot" (describing real-time code suggestions) and "Our response to the war in Ukraine" (mentioning GitHub's support for Ukraine). At the bottom right, there is a "Latest changes" section.

The screenshot shows the GitHub homepage with a user profile dropdown menu open. The menu includes options like 'Get suggestions for lines and entire functions real-time', 'Your profile', 'Your repositories', 'Your codespaces', 'Your projects', 'Your stars', 'Your gists', 'Upgrade', 'Feature preview', 'Help', 'Settings', and 'Sign out'. The 'Your repositories' option is highlighted with a red oval.

Go to your repositories

The screenshot shows the GitHub repositories page for the user 'kroykishan8'. The user has a large profile picture with a red GitHub logo. The page title is 'kroykishan8'. The top navigation bar includes 'Overview', 'Repositories' (which is selected), 'Projects', 'Packages', and 'Stars'. Below the navigation is a search bar with 'Find a repository...' and filters for 'Type', 'Language', 'Sort', and a 'New' button. The main content area displays the message 'kroykishan8 doesn't have any public repositories yet.'

LAB:

1. Go to mumbai ec2-instance and run following commands

```
sudo su
ls
which git
mkdir mumbaigit
ls
cd mumbaigit
git init
cat > mumbaofile1
```

Write some line in mumbaofile1 and then press enter and do ctrl+d to save.

```
ec2-user@ip-172-31-15-53:~
```

```
Using username "ec2-user".  
Authenticating with public key "imported-openssh-key"  
Last login: Fri Jul  8 07:05:34 2022 from 203.189.248.184
```

```
 _ | ( _ / )  
 _ | \ _ | _ |  
Amazon Linux 2 AMI
```

```
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-172-31-15-53 ~]$
```

```
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-172-31-15-53 ~]$ sudo su root  
[root@ip-172-31-15-53 ec2-user]# ls  
[root@ip-172-31-15-53 ec2-user]# which git  
/bin/git  
[root@ip-172-31-15-53 ec2-user]# mkdir mumbaigit  
[root@ip-172-31-15-53 ec2-user]# ls  
mumbaigit  
[root@ip-172-31-15-53 ec2-user]# cd mumbaigit  
[root@ip-172-31-15-53 mumbaigit]# git init  
hint: Using 'master' as the name for the initial branch. This default branch name  
hint: is subject to change. To configure the initial branch name to use in all  
hint: of your new repositories, which will suppress this warning, call:  
hint:  
hint:   git config --global init.defaultBranch <name>  
hint:  
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and  
hint: 'development'. The just-created branch can be renamed via this command:  
hint:  
hint:   git branch -m <name>  
Initialized empty Git repository in /home/ec2-user/mumbaigit/.git/  
[root@ip-172-31-15-53 mumbaigit]#  
[root@ip-172-31-15-53 mumbaigit]# ls  
[root@ip-172-31-15-53 mumbaigit]# cat > mumbaifile1  
This is first line of code  
This is second line of code  
[root@ip-172-31-15-53 mumbaigit]# cat mumbaifile1  
This is first line of code  
This is second line of code  
[root@ip-172-31-15-53 mumbaigit]#
```

```
git status  
git add .  
git commit -m "adding first file from mumbai"
```

```
[root@ip-172-31-15-53 mumbaigit]# git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    mumbaifile1

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-15-53 mumbaigit]# 
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   mumbaifile1

[root@ip-172-31-15-53 mumbaigit]# 
[root@ip-172-31-15-53 mumbaigit]# git commit -m "adding first file from mumbai"
[master (root-commit) 0e106e4] adding first file from mumbai
1 file changed, 2 insertions(+)
create mode 100644 mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git log
git log --oneline
```

```
[root@ip-172-31-15-53 mumbaigit]# git log
commit 0e106e443bed2cf34eee5df9e823ccfd36976610 (HEAD -> master)
Author: mumbaiemp <mumbaiemp@gmail.com>
Date:   Fri Jul 8 07:47:08 2022 +0000

    adding first file from mumbai
[root@ip-172-31-15-53 mumbaigit]# git log --oneline
0e106e4 (HEAD -> master) adding first file from mumbai
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git show <commid-id>
```

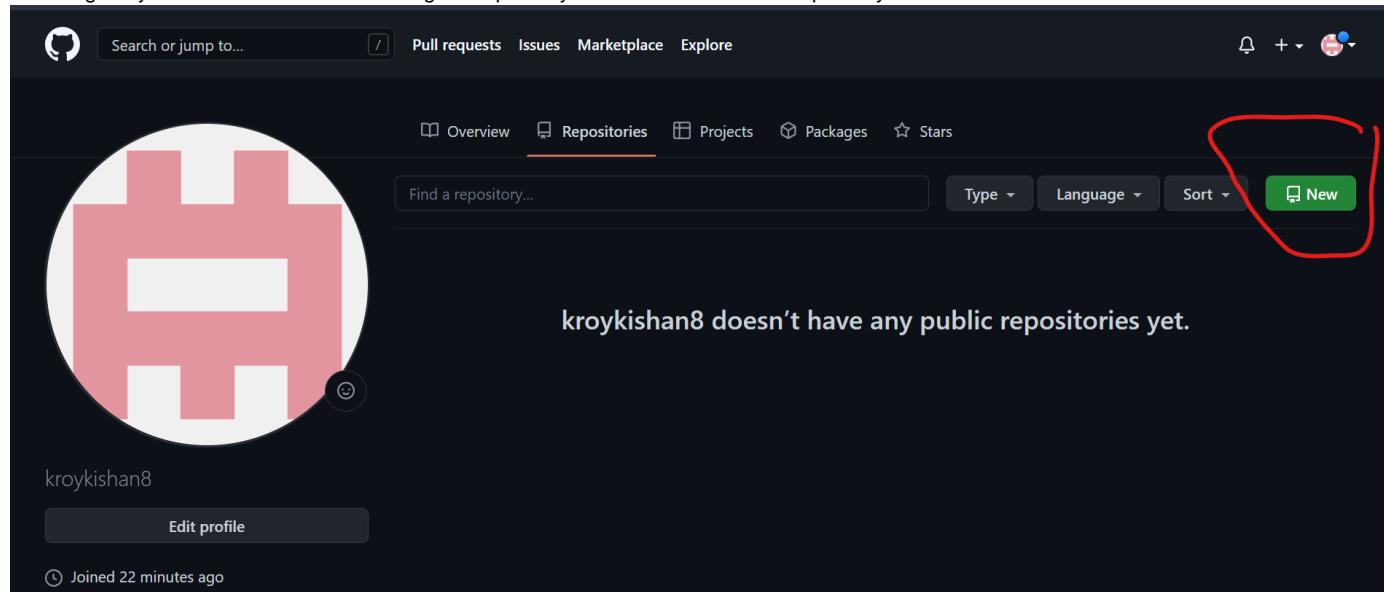
```
[root@ip-172-31-15-53 mumbaigit]# git log --oneline
0e106e4 (HEAD -> master) adding first file from mumbai
[root@ip-172-31-15-53 mumbaigit]# git show 0e106e4
commit 0e106e443bed2cf34eee5df9e823ccfd36976610 (HEAD -> master)
Author: mumbaiemp <mumbaiemp@gmail.com>
Date:   Fri Jul 8 07:47:08 2022 +0000

    adding first file from mumbai

diff --git a/mumbaifile1 b/mumbaifile1
new file mode 100644
index 0000000..21dfdb2
--- /dev/null
+++ b/mumbaifile1
@@ -0,0 +1,2 @@
+This is first line of code
+This is second line of code
[root@ip-172-31-15-53 mumbaigit]#
```

Now we have to push this code to Github repo (central repository)

For this go to your Github account and then go to repository and then create a new repository.



Search or jump to... / Pull requests Issues Marketplace Explore

Overview Repositories Projects Packages Stars

Find a repository... Type Language Sort

New

kroykishan8 doesn't have any public repositories yet.

kroykishan8

Edit profile

Joined 22 minutes ago

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *

 kroykishan8 ▾

Repository name *

MycentralRepo



Great repository names are short and memorable. Need inspiration? How about [literate-winner](#)?

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.](#)

 **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: **None ▾**

 You are creating a public repository in your personal account.

Create repository

kroykishan8 / MycentralRepo Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

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

Gitpod Set up in Desktop or HTTPS SSH <https://github.com/kroykishan8/MycentralRepo.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 "# MycentralRepo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/kroykishan8/MycentralRepo.git
git push -u origin main
```

Now copy the following command circled with red color

kroykishan8 / MycentralRepo Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

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

Gitpod Set up in Desktop or HTTPS SSH <https://github.com/kroykishan8/MycentralRepo.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 "# MycentralRepo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/kroykishan8/MycentralRepo.git
git push -u origin main
```

Now paste the code in Mumbai ec2-instance

```
+This is second line of code
[root@ip-172-31-15-53 mumbaigit]# git remote add origin https://github.com/kroykishan8/MycentralRepo.git
[root@ip-172-31-15-53 mumbaigit]#
```

```
git push origin
```

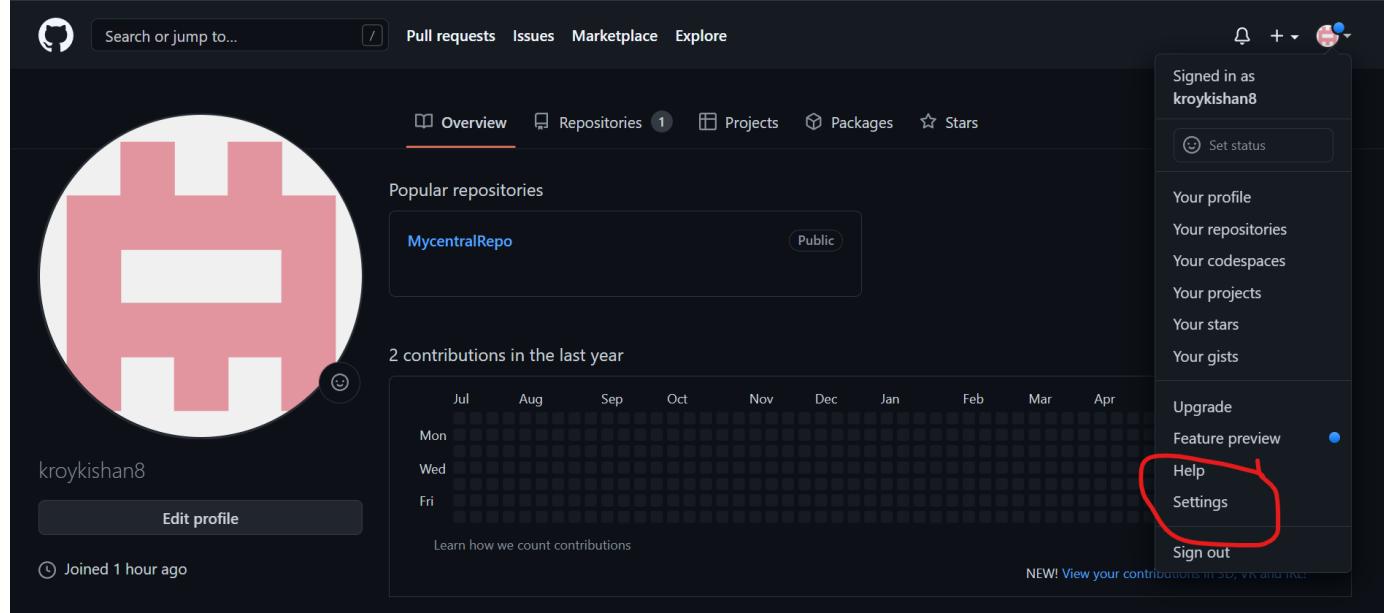
```
#Currently our main branch is master so they will ask to run this code#
#git push --set-upstream origin master
#after this it will ask for username and password of central github
account
#if you provide username and password it will again fail
#in place of password it will ask for a token so we will create token
# will run this command again "git push --set-upstream origin master"
#in place of password we will replace that with token.
```

```
[root@ip-172-31-15-53 mumbaigit]# git push origin
fatal: The current branch master has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin master

[root@ip-172-31-15-53 mumbaigit]# git push --set-upstream origin master
Username for 'https://github.com': kroykishan8
Password for 'https://kroykishan8@github.com':
remote: Support for password authentication was removed on August 13, 2021. Please use a personal access token instead.
remote: Please see https://github.blog/2020-12-15-token-authentication-requirements-for-git-operations/ for more information.
fatal: Authentication failed for 'https://github.com/kroykishan8/MycentralRepo.git'
[root@ip-172-31-15-53 mumbaigit]#
```

Now go to Github account to generate a token



The screenshot shows a GitHub profile page for the user 'kroykishan8'. The profile picture is a large red and white geometric shape. The user has 2 contributions in the last year. The 'Settings' option in the top right corner of the sidebar is circled in red.

A screenshot of the GitHub Public profile page for the user 'kroykishan8'. The top navigation bar includes links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The user's profile picture is a pink 'H' icon. The profile summary shows 'Your personal account'. On the left, a sidebar lists 'Public profile' (selected), 'Account', 'Appearance', 'Accessibility', 'Notifications', 'Access', 'Billing and plans', 'Emails', 'Password and authentication', 'SSH and GPG keys', 'Organizations', and 'Moderation'. The main content area is titled 'Public profile' and contains sections for 'Name' (input field), 'Profile picture' (a large circular image of the 'H' icon with an 'Edit' button), 'Public email' (a dropdown menu with 'Select a verified email to display'), and 'Bio' (a text input field with placeholder 'Tell us a little bit about yourself'). A note at the bottom of the bio section says 'You can @mention other users and organizations to link to them.'

The image shows two screenshots of the GitHub web interface. The top screenshot is the 'Company' section of the profile settings, featuring fields for company and location, an 'Update profile' button, and a note about optional data sharing. The bottom screenshot is the 'Developer settings' page, showing sections for GitHub Apps, OAuth Apps, and Personal access tokens, with a note about building GitHub Apps. Red ovals highlight the 'Developer settings' link in the profile sidebar and the 'Personal access tokens' link in the developer settings sidebar.

Company

You can @mention your company's GitHub organization to link it.

Location

All of the fields on this page are optional and can be deleted at any time, and by filling them out, you're giving us consent to share this data wherever your user profile appears. Please see our [privacy statement](#) to learn more about how we use this information.

[Update profile](#)

Contributions & Activity

Make profile private and hide activity Beta
Enabling this will hide your contributions and activity from your GitHub profile and from social features like followers, stars, feeds, leaderboards and releases.

Include private contributions on my profile
Your contribution graph, achievements, and activity overview will show your private contributions without revealing any repository or

Search or jump to... / Pull requests Issues Marketplace Explore

Settings / Developer settings

GitHub Apps

New GitHub App

GitHub Apps

Want to build something that integrates with and extends GitHub? [Register a new GitHub App](#) to get started developing on the GitHub API. You can also read more about building GitHub Apps in our [developer documentation](#).

GitHub Apps

Personal access tokens

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

Settings / Developer settings

GitHub Apps

OAuth Apps

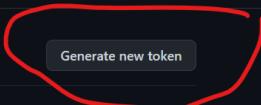
Personal access tokens

Personal access tokens

Need an API token for scripts or testing? [Generate a personal access token](#) for quick access to the [GitHub API](#).

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About



GitHub Apps

OAuth Apps

Personal access tokens

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

CentralReposit

What's this token for?

Expiration *

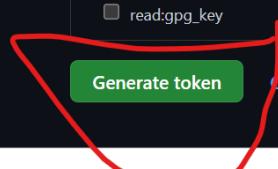
90 days The token will expire on Thu, Oct 6 2022

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo:deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<hr/>	
<input type="checkbox"/> workflow	Update GitHub Action workflows
<input type="checkbox"/> write:discussion	Read and write team discussions
<input type="checkbox"/> read:discussion	Read team discussions
<hr/>	
<input type="checkbox"/> admin:enterprise	Full control of enterprises
<input type="checkbox"/> manage_runners:enterprise	Manage enterprise runners and runner-groups
<input type="checkbox"/> manage_billing:enterprise	Read and write enterprise billing data
<input type="checkbox"/> read:enterprise	Read enterprise profile data
<hr/>	
<input type="checkbox"/> project	Full control of projects
<input type="checkbox"/> read:project	Read access of projects
<hr/>	
<input type="checkbox"/> admin:gpg_key	Full control of public user GPG keys
<input type="checkbox"/> write:gpg_key	Write public user GPG keys
<input type="checkbox"/> read:gpg_key	Read public user GPG keys

Generate token **Cancel**



Some of the scopes you've selected are included in other scopes. Only the minimum set of necessary scopes has been saved.

Settings / Developer settings

GitHub Apps

OAuth Apps

Personal access tokens

Personal access tokens

Generate new token

Revoke all

Tokens you have generated that can be used to access the GitHub API.

Make sure to copy your personal access token now. You won't be able to see it again!

ghp_sCyK5CeB... [Copy] [Delete]

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

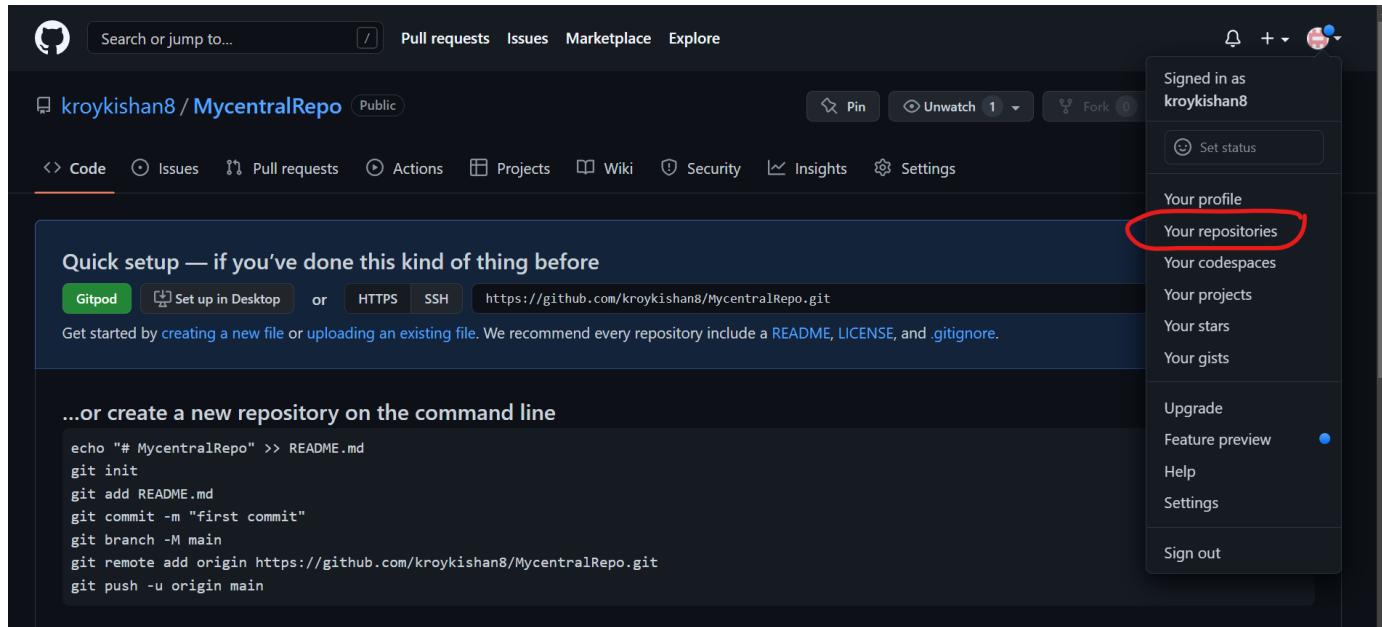
Note: *Copy this token and keep it safely in some file or document. Because you will not be able to see this token value again. If token is lost then you have to generate new token again.*

Now again come to Mumbai ec2-instance and run following command again

```
git push --set-upstream origin master
#username: give username
#password: copy the token value what we created
```

```
[root@ip-172-31-15-53 mumbaigit]# git push --set-upstream origin master
Username for 'https://github.com': kroykishan8
Password for 'https://kroykishan8@github.com': [REDACTED]
[root@ip-172-31-15-53 mumbaigit]# git push --set-upstream origin master
Username for 'https://github.com': kroykishan8
Password for 'https://kroykishan8@github.com': [REDACTED]
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 255 bytes | 255.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/kroykishan8/MycentralRepo.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
[root@ip-172-31-15-53 mumbaigit]# [REDACTED]
```

Now again go to Github account and then open your repository.



Signed in as **kroykishan8**

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

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

Gitpod Set up in Desktop or HTTPS SSH <https://github.com/kroykishan8/MycentralRepo.git>

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

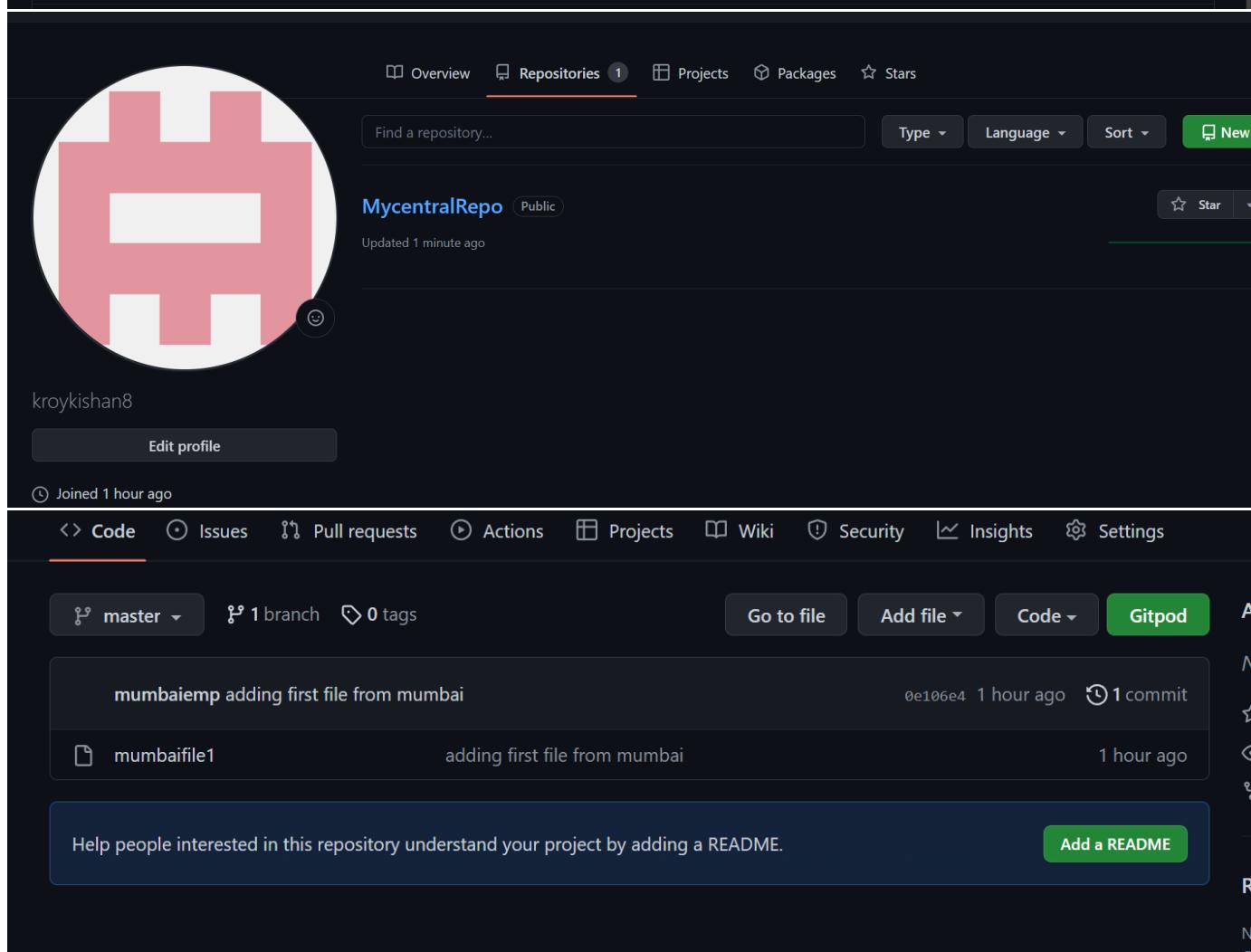
...or create a new repository on the command line

```
echo "# MycentralRepo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/kroykishan8/MycentralRepo.git
git push -u origin main
```

Your profile Your repositories (circled in red)

Your codespaces Your projects Your stars Your gists

Upgrade Feature preview Help Settings Sign out



Overview Repositories 1 Projects Packages Stars

Find a repository... Type Language Sort New

MycentralRepo Public Updated 1 minute ago Star

kroykishan8 Edit profile

Joined 1 hour ago

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

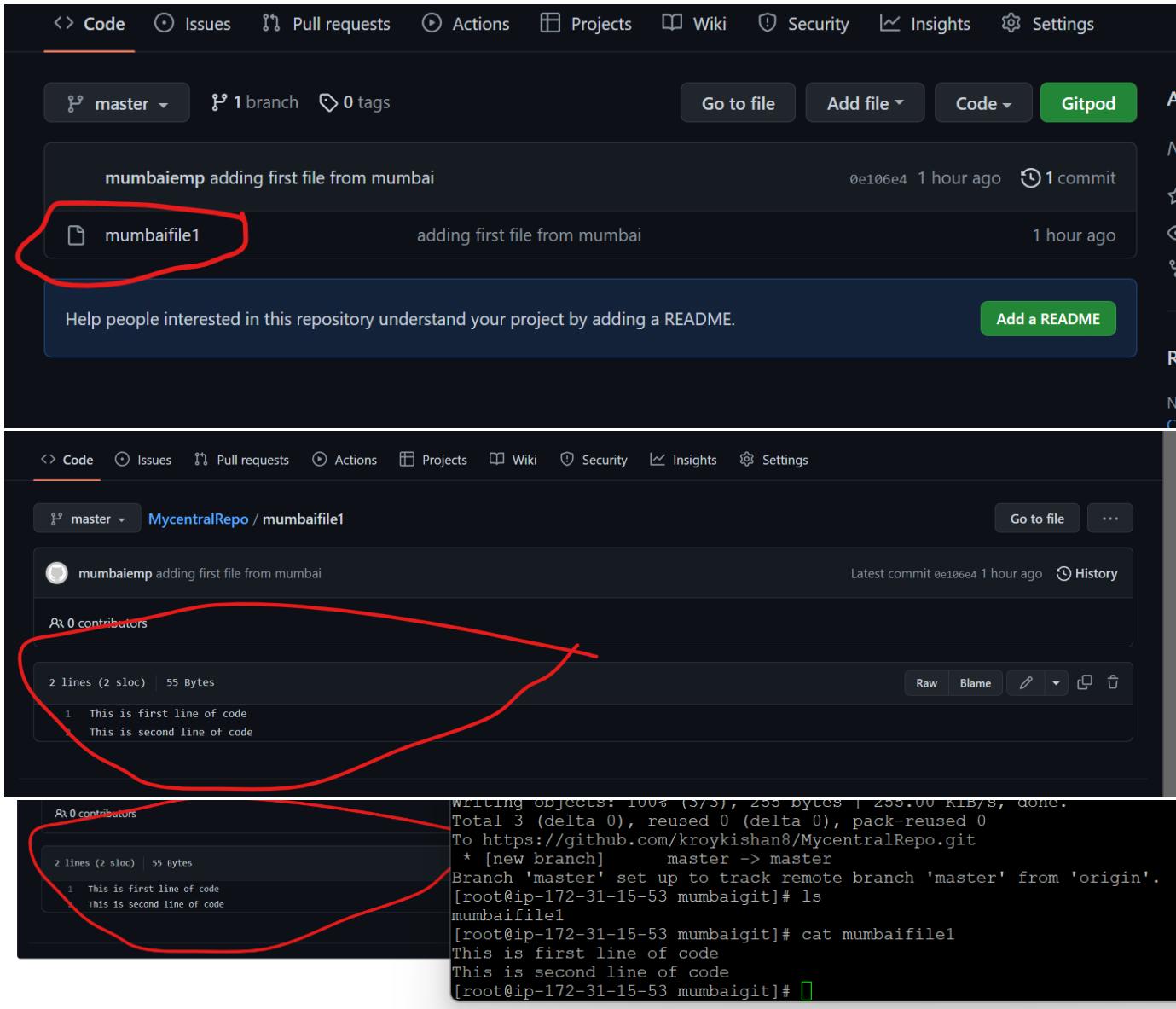
master 1 branch 0 tags Go to file Add file Code Gitpod

mumbaiemp adding first file from mumbai 0e106e4 1 hour ago 1 commit

mumbaifile1 adding first file from mumbai 1 hour ago

Help people interested in this repository understand your project by adding a README. Add a README

We can see our mumbaifile1 uploaded to centralgithubrepo.



Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags Go to file Add file Code Gitpod

mumbaiemp adding first file from mumbai 0e106e4 1 hour ago 1 commit

mumbaifile1 adding first file from mumbai 1 hour ago

Help people interested in this repository understand your project by adding a README. Add a README

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master MycentralRepo / mumbaifile1 Go to file ...

mumbaiemp adding first file from mumbai Latest commit 0e106e4 1 hour ago History

0 contributors

2 lines (2 sloc) | 55 Bytes

1 This is first line of code
2 This is second line of code

Raw Blame

0 contributors

2 lines (2 sloc) | 55 bytes

1 This is first line of code
2 This is second line of code

```
Writing objects: 100% (3/3), 255 bytes | 255.00 KIB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/kroykishan8/MycentralRepo.git
 * [new branch] master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
[root@ip-172-31-15-53 mumbaigit]# ls
mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# cat mumbaifile1
This is first line of code
This is second line of code
[root@ip-172-31-15-53 mumbaigit]#
```

Lets update one more line of code in that file.

```
cat > mumbaifile1
git status
git add .
git status
git commit -m "Latest commit from mumbai ec2-linux"
git log
git show <commit-id>
git push
#username
#password: token paste here
```

```
[root@ip-172-31-15-53 mumbaigit]# cat >> mumbaifile1
This is my third line of code from mumbai linux
[root@ip-172-31-15-53 mumbaigit]# git status
On branch master
Your branch is up to date with 'origin/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:   mumbaifile1

no changes added to commit (use "git add" and/or "git commit -a")
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# 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)
    modified:   mumbaifile1

[root@ip-172-31-15-53 mumbaigit]# 
[root@ip-172-31-15-53 mumbaigit]# git commit -m "Latest commit from mumbai ec2-l
inux"
[master 36ea66d] Latest commit from mumbai ec2-linux
1 file changed, 1 insertion(+)
[root@ip-172-31-15-53 mumbaigit]# 
[root@ip-172-31-15-53 mumbaigit]# git log
commit 36ea66da885ae75722ac8bcdde89f1e563d3b093 (HEAD -> master)
Author: mumbaiemp <mumbaiemp@gmail.com>
Date:   Fri Jul 8 09:04:11 2022 +0000

    Latest commit from mumbai ec2-linux

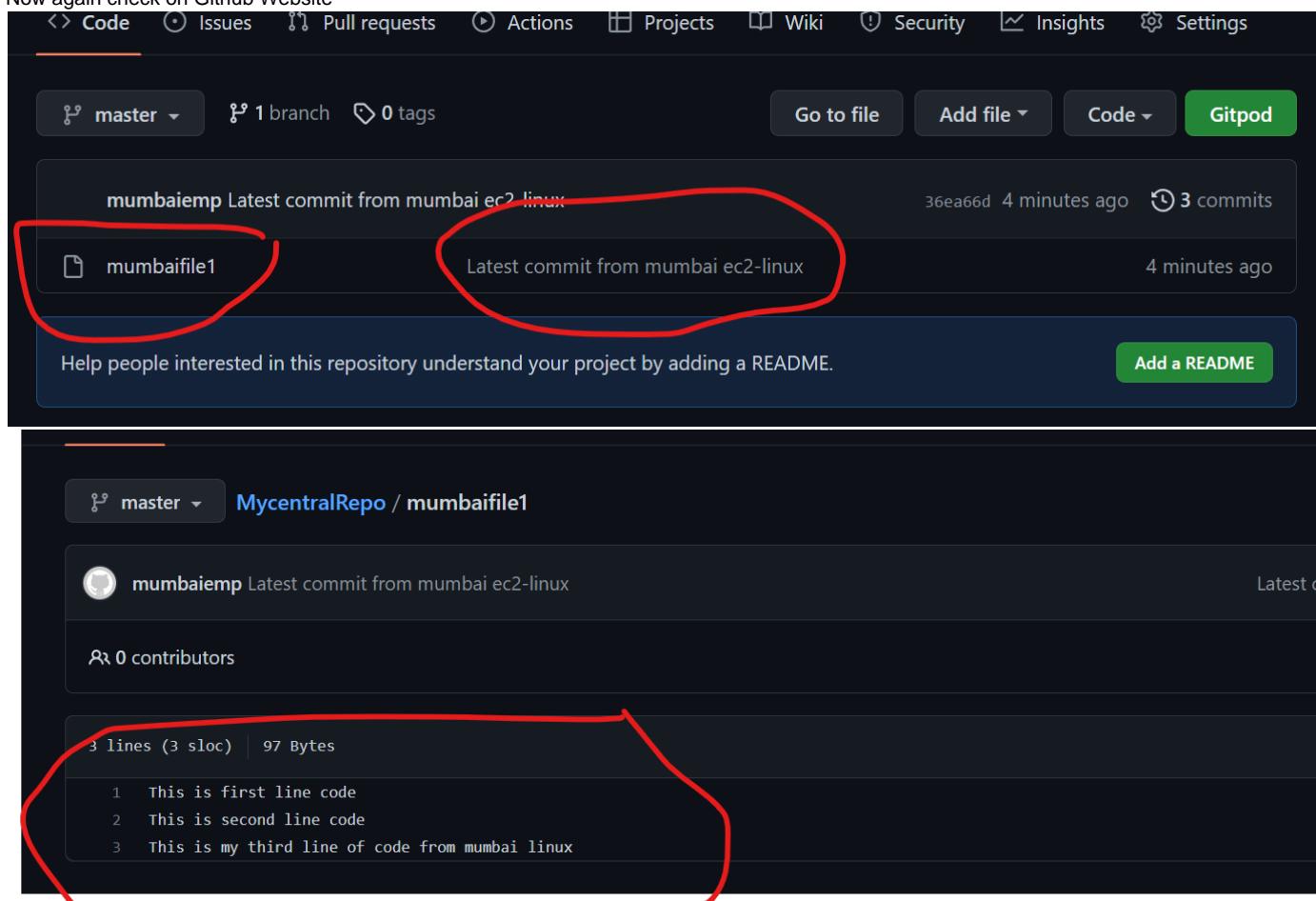
[root@ip-172-31-15-53 mumbaigit]# git show 36ea66da
commit 36ea66da885ae75722ac8bcdde89f1e563d3b093 (HEAD -> master)
Author: mumbaiemp <mumbaiemp@gmail.com>
Date:   Fri Jul 8 09:04:11 2022 +0000

    Latest commit from mumbai ec2-linux

diff --git a/mumbaifile1 b/mumbaifile1
index 4990907..485123f 100644
--- a/mumbaifile1
+++ b/mumbaifile1
@@ -1,2 +1,3 @@
 This is first line code
 This is second line code
+This is my third line of code from mumbai linux
[root@ip-172-31-15-53 mumbaigit]# 
```

```
[root@ip-172-31-15-53 mumbaigit]# git push
Username for 'https://github.com': kroykishan8
Password for 'https://kroykishan8@github.com':
remote: Invalid username or password.
fatal: Authentication failed for 'https://github.com/kroykishan8/MycentralRepo.git'
[root@ip-172-31-15-53 mumbaigit]# git push
Username for 'https://github.com': kroykishan8
Password for 'https://kroykishan8@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 318 bytes | 318.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/kroykishan8/MycentralRepo.git
  895a6b3..36ea66d master -> master
[root@ip-172-31-15-53 mumbaigit]#
```

Now again check on Github Website



Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags Go to file Add file Code Gitpod

mumbaient Latest commit from mumbai ec2-linux 36ea66d 4 minutes ago 3 commits

mumbafile1 Latest commit from mumbai ec2-linux 4 minutes ago

Help people interested in this repository understand your project by adding a README. Add a README

master MycentralRepo / mumbafile1

mumbaient Latest commit from mumbai ec2-linux

0 contributors

3 lines (3 sloc) 97 Bytes

```
1 This is first line code
2 This is second line code
3 This is my third line of code from mumbai linux
```

The screenshot shows a GitHub commit page for a repository named 'MycentralRepo / mumbaifile1'. The commit was made by 'mumbaiemp' on Jul 8, 2022. The commit message is 'Latest commit from mumbai ec2-linux'. The commit has 3 lines (3 sloc) and 97 Bytes. The code content is:

```
1 This is first line code
2 This is second line code
3 This is my third line of code from mumbai linux
```

The 'History' button in the top right corner is circled in red. Below the commit, the 'History' section shows the same commit and a previous commit from the same author. The previous commit is 'first commit mumbai' with commit message 'mumbaiemp committed 7 minutes ago'. The commit details are:

Latest commit from mumbai ec2-linux
mumbaiemp committed 5 minutes ago

first commit mumbai
mumbaiemp committed 7 minutes ago

The commit details for the latest commit are:

Latest commit from mumbai ec2-linux
mumbaiemp committed 5 minutes ago

Showing 1 changed file with 1 addition and 0 deletions.

The diff view shows the changes:

```
1 This is first line code
2 This is second line code
3 + This is my third line of code from mumbai linux
```

Red boxes highlight the 'History' button and the commit details for the latest commit.

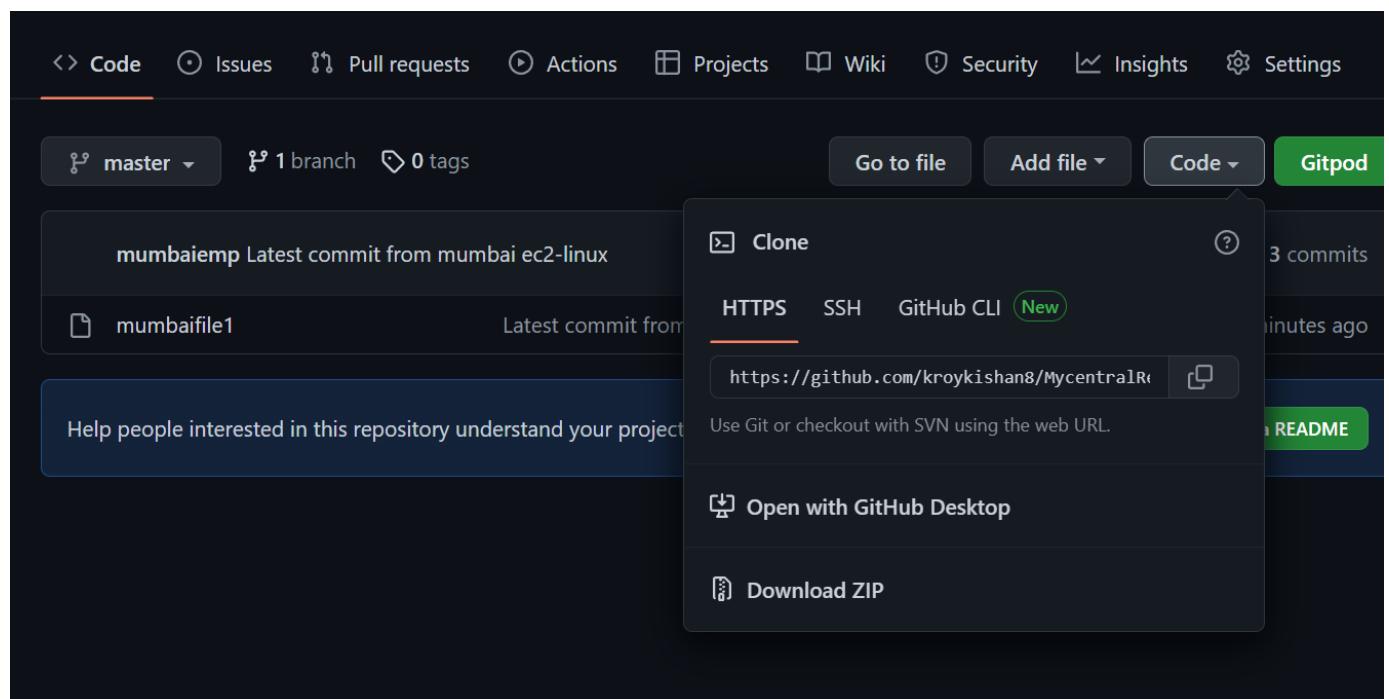
Now Go to N.virginia ec2-instance and run following command.

```
sudo su root
mkdir usagit
ls
cd usagit
git init
```

```
Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-28-241 ~]$ sudo su root
[root@ip-172-31-28-241 ec2-user]# ls
[root@ip-172-31-28-241 ec2-user]# mkdir usagit
[root@ip-172-31-28-241 ec2-user]# ls
usagit
[root@ip-172-31-28-241 ec2-user]# cd usagit
[root@ip-172-31-28-241 usagit]# git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ec2-user/usagit/.git/
[root@ip-172-31-28-241 usagit]#
```

```
git remote add origin <github central repo url>
#copy from github
```



The screenshot shows a GitHub repository page for a user named 'mumbailemp'. The page displays basic repository statistics: 1 branch and 0 tags. A prominent 'Clone' button is visible, with a dropdown menu open. The menu includes options for 'HTTPS', 'SSH', and 'GitHub CLI'. The 'HTTPS' option is selected and highlighted with a green 'New' badge. Below these options is the URL <https://github.com/kroykishan8/MycentralRe>. The dropdown also contains links for 'Open with GitHub Desktop' and 'Download ZIP'.

```
initialized empty git repository in /home/usager/usagit/.git  
[root@ip-172-31-28-241 usagit]# git remote add origin https://github.com/kroykishan8/MycentralRepo.git  
[root@ip-172-31-28-241 usagit]# ls  
[root@ip-172-31-28-241 usagit]# ls  
[root@ip-172-31-28-241 usagit]# 
```

```
git log  
git status
```

```
[root@ip-172-31-28-241 usagit]# git log  
fatal: your current branch 'master' does not have any commits yet  
[root@ip-172-31-28-241 usagit]# git status  
On branch master  
  
No commits yet  
  
nothing to commit (create/copy files and use "git add" to track)  
[root@ip-172-31-28-241 usagit]# 
```

```
git pull origin master  
ls  
cat mumbaifile1  
#will show the same file content in north virginia also
```

```
[root@ip-172-31-28-241 usagit]# git pull origin master  
remote: Enumerating objects: 9, done.  
remote: Counting objects: 100% (9/9), done.  
remote: Compressing objects: 100% (5/5), done.  
remote: Total 9 (delta 0), reused 9 (delta 0), pack-reused 0  
Unpacking objects: 100% (9/9), 762 bytes | 762.00 KiB/s, done.  
From https://github.com/kroykishan8/MycentralRepo  
 * branch master      -> FETCH_HEAD  
 * [new branch] master -> origin/master  
[root@ip-172-31-28-241 usagit]# ls
```

```
mumbaifile1  
[root@ip-172-31-28-241 usagit]# ls  
mumbaifile1  
[root@ip-172-31-28-241 usagit]# cat mumbaifile1  
This is first line code  
This is second line code  
This is my third line of code from mumbai linux  
[root@ip-172-31-28-241 usagit]# 
```

Now try add some more line from N.virginia ec2-instance as us employee

```
cat >> mumbaifile1
git status
git add .
git status
git commit -m "first commit by us employee"
```

```
[root@ip-172-31-28-241 usagit]# cat >> mumbaifile1
This is fourth line by us employee
This is fifth line by us employee
[root@ip-172-31-28-241 usagit]# 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:   mumbaifile1

no changes added to commit (use "git add" and/or "git commit -a")
[root@ip-172-31-28-241 usagit]# git add .
[root@ip-172-31-28-241 usagit]# git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   mumbaifile1

[root@ip-172-31-28-241 usagit]# git commit -m " first commit by us employee"
[master 36315d2] first commit by us employee
 1 file changed, 2 insertions(+)
[root@ip-172-31-28-241 usagit]# 
```

```
git push --set-upstream origin master
#username
#password: paste same generated token
```

```
[root@ip-172-31-28-241 usagit]# git push --set-upstream origin master
Username for 'https://github.com': kroykishan8
Password for 'https://kroykishan8@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 332 bytes | 332.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/kroykishan8/MycentralRepo.git
  36ea66d..36315d2 master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
[root@ip-172-31-28-241 usagit]# 
```

```
git log
git log --oneline
git show <commit-id>
```

```
[root@ip-172-31-28-241 usagit]# git log
commit 36315d2619f0d638c3325d3b6bc80baccb04c4ae (HEAD -> master, origin/master)
Author: usemp <usemp@gmail.com>
Date:   Fri Jul 8 09:24:19 2022 +0000

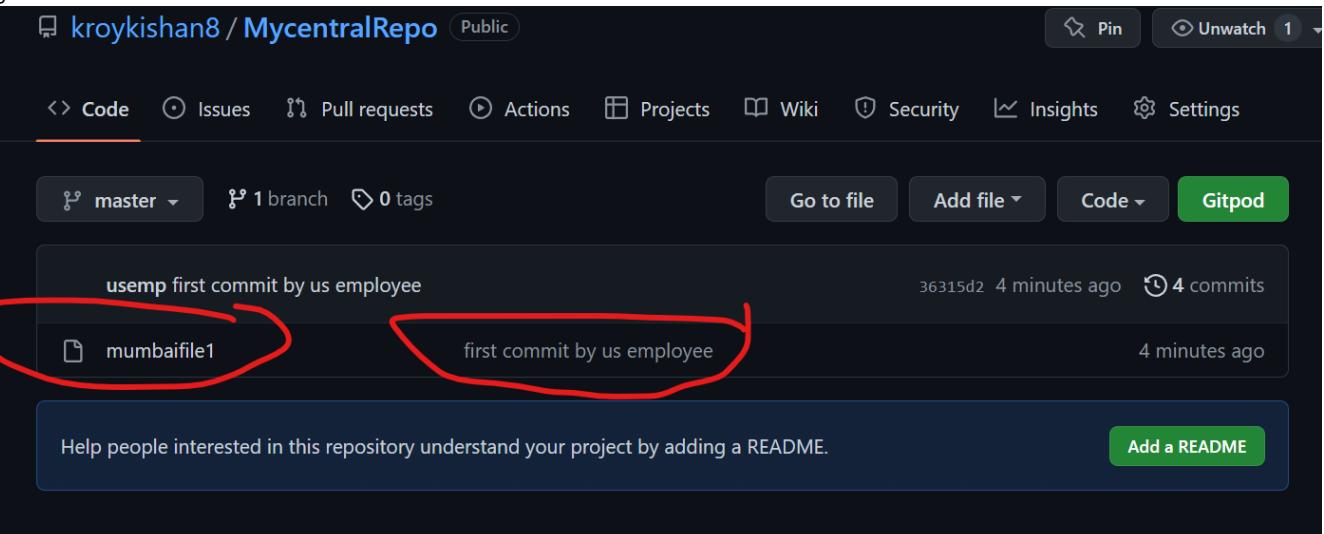
    first commit by us employee

[root@ip-172-31-28-241 usagit]# git log --oneline
36315d2 (HEAD -> master, origin/master) first commit by us employee
36ea66d Latest commit from mumbai ec2-linux
895a6b3 first commit mumbai
0e106e4 adding first file from mumbai
[root@ip-172-31-28-241 usagit]# git show 36315d2
commit 36315d2619f0d638c3325d3b6bc80baccb04c4ae (HEAD -> master, origin/master)
Author: usemp <usemp@gmail.com>
Date:   Fri Jul 8 09:24:19 2022 +0000

    first commit by us employee

diff --git a/mumbaifile1 b/mumbaifile1
index 485123f..d5b2823 100644
--- a/mumbaifile1
+++ b/mumbaifile1
@@ -1,3 +1,5 @@
This is first line code
This is second line code
This is my third line of code from mumbai linux
+This is fourth line by us employee
+This is fifth line by us employee
[root@ip-172-31-28-241 usagit]#
```

Now go to Github account and check their



kroykishan8 / MycentralRepo Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags Go to file Add file Code Gitpod

usemp first commit by us employee 36315d2 4 minutes ago 4 commits

mumbaifile1 4 minutes ago

first commit by us employee 4 minutes ago

Help people interested in this repository understand your project by adding a README. Add a README

master  MycentralRepo / mumbaifile1

 **usemp** first commit by us employee

 0 contributors

5 lines (5 sloc) | 166 Bytes

```
1 This is first line code
2 This is second line code
3 This is my third line of code from mumbai linux
4 This is fourth line by us employee
5 This is fifth line by us employee
```

Showing 1 changed file with 2 additions and 0 deletions.

2 mumbaifile1

```
@@ -1,3 +1,5 @@
... ...
1 1 This is first line code
2 2 This is second line code
3 3 This is my third line of code from mumbai linux
4 + This is fourth line by us employee
5 + This is fifth line by us employee
```

Now us employee pushed the code, now next day again mumbai employee will login to his ec2-instance, and to check what work was done by us employee he will run following command.

```
sudo su root
ls
cd mumbaigit
ls
git log
```

```
[ec2-user@ip-172-31-15-53 ~]$ sudo su root
[root@ip-172-31-15-53 ec2-user]# ls
mumbaigit
[root@ip-172-31-15-53 ec2-user]# cd mumbaigit/
[root@ip-172-31-15-53 mumbaigit]# ls
mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# git log
commit 36ea66da885ae75722ac8bcdde89f1e563d3b093 (HEAD -> master, origin/master)
Author: mumbaiemp <mumbaiemp@gmail.com>
Date:   Fri Jul 8 09:04:11 2022 +0000

    Latest commit from mumbai ec2-linux

commit 895a6b3ca75cf0ff0dde6e5443b452bdf3250c81
```

```
git push origin master
cat mumbaifile1
```

```
[root@ip-172-31-15-53 mumbaigit]# git pull origin master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 312 bytes | 312.00 KiB/s, done.
From https://github.com/kroykishan8/MycentralRepo
 * branch            master      -> FETCH_HEAD
   36ea66d..36315d2  master      -> origin/master
Updating 36ea66d..36315d2
Fast-forward
  mumbaifile1 | 2 ++
  1 file changed, 2 insertions(+)
[root@ip-172-31-15-53 mumbaigit]#
```

```
mumbaifile1 | 2 ++
1 file changed, 2 insertions(+)
[root@ip-172-31-15-53 mumbaigit]# cat mumbaifile1
This is first line code
This is second line code
This is my third line of code from mumbai linux
This is fourth line by us employee
This is fifth line by us employee
[root@ip-172-31-15-53 mumbaigit]#
```

Now again Mumbai employee will add some line and he will push, so that we can continue this process.

To ignore Some file run these command.

```
vi .gitignore
ls -a
git add .gitignore
git status
git commit -m "ignore file format"
```

```
[root@ip-172-31-15-53 mumbaigit]# cat .gitignore
```

```
*.css
*.txt
```

```
[root@ip-172-31-15-53 mumbaigit]# vi .gitignore
[root@ip-172-31-15-53 mumbaigit]# ls -l
total 4
-rw-r--r-- 1 root root 166 Jul  8 09:37 mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# ls -a
.  ..  .git  .gitignore  mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# git add .gitignore
[root@ip-172-31-15-53 mumbaigit]# 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:   .gitignore
```

```
[root@ip-172-31-15-53 mumbaigit]# git commit -m "ignore file format"
[master 1843568] ignore file format
 1 file changed, 2 insertions(+)
 create mode 100644 .gitignore
```

```
touch a.css b.css a.txt abc.java kroy.html
```

```

root@ip-172-31-15-53 mumbaigit]# touch a.css b.css a.txt abc.java kroy.html
root@ip-172-31-15-53 mumbaigit]# ls
abc.java a.css a.txt b.css kroy.html mumbaifile1
root@ip-172-31-15-53 mumbaigit]# git add .
root@ip-172-31-15-53 mumbaigit]# git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   abc.java
    new file:   kroy.html

root@ip-172-31-15-53 mumbaigit]# git commit -m "ignore css and txt file"
[master 9d17a62] ignore css and txt file
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 abc.java
create mode 100644 kroy.html

```

```

git push
#username
#password:token

```

```

[root@ip-172-31-15-53 mumbaigit]# git push
Username for 'https://github.com': kroykishan8
Password for 'https://kroykishan8@github.com':
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 515 bytes | 515.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/kroykishan8/MycentralRepo.git
  36315d2..9d17a62  master -> master
[root@ip-172-31-15-53 mumbaigit]# 

```

The screenshot shows a GitHub repository page for a user named 'mumbaigit'. The repository has a single branch named 'master' and 0 tags. The commit history shows a single commit from '9d17a62' made 7 minutes ago, with 6 commits. The commit message is 'ignore css and txt file'. The commit details show four files: '.gitignore', 'abc.java', 'kroy.html', and 'mumbaifile1'. The '.gitignore' file is an 'ignore file format', 'abc.java' is 'ignore css and txt file', 'kroy.html' is 'ignore css and txt file', and 'mumbaifile1' is 'first commit by us employee'. The repository has no description, website, stars, watchers, or forks. It also has no releases published.

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags Go to file Add file Code Gitpod

mumbaigit ignore css and txt file 9d17a62 7 minutes ago 6 commits

.gitignore ignore file format 9 minutes ago

abc.java ignore css and txt file 7 minutes ago

kroy.html ignore css and txt file 7 minutes ago

mumbaifile1 first commit by us employee 5 hours ago

No description, website

0 stars 1 watching 0 forks

Releases

No releases published Create a new release

Help people interested in this repository understand your project by adding a README. Add a README

Note: create file .css and check git status and then again create a file .html and then check git status

```
[root@ip-172-31-15-53 mumbaigit]# touch abc.css
[root@ip-172-31-15-53 mumbaigit]# git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
[root@ip-172-31-15-53 mumbaigit]# touch ab.html
[root@ip-172-31-15-53 mumbaigit]# 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)
    ab.html

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-15-53 mumbaigit]#
```

```
git log --grep "ignore"
```

```
[root@ip-172-31-15-53 mumbaigit]# git log --grep "ignore"
commit 9d17a6263adce930524ee9feddb55927393c23b1 (HEAD -> master, origin/master)
Author: mumbaiemp <mumbaiemp@gmail.com>
Date:   Fri Jul 8 14:45:50 2022 +0000

    ignore css and txt file

commit 1843568116f8672fd71ae4fec96c6a17dd660854
Author: mumbaiemp <mumbaiemp@gmail.com>
Date:   Fri Jul 8 14:43:40 2022 +0000

    ignore file format
[root@ip-172-31-15-53 mumbaigit]#
```

BRANCH

```
git log --oneline
git branch
```

```
[root@ip-172-31-15-53 mumbaigit]# git log --oneline
9d17a62 (HEAD -> master, origin/master) ignore css and txt file
1843568 ignore file format
36315d2 first commit by us employee
36ea66d Latest commit from mumbai ec2-linux
895a6b3 first commit mumbai
0e106e4 adding first file from mumbai
[root@ip-172-31-15-53 mumbaigit]# git branch
* master
[root@ip-172-31-15-53 mumbaigit]#
```

```
git branch branch1
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
* master
[root@ip-172-31-15-53 mumbaigit]# git branch branch1
[root@ip-172-31-15-53 mumbaigit]# git branch
  branch1
* master
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git branch
ls
git checkout branch1
ls
#all the files available in master will be present in branch 1 also
git branch
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
  branch1
* master
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  kroy.html  mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# git checkout branch1
Switched to branch 'branch1'
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  kroy.html  mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# git branch
* branch1
  master
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git branch
git log --oneline
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
* branch1
  master
[root@ip-172-31-15-53 mumbaigit]# git log --oneline
9d17a62 (HEAD -> branch1, origin/master, master) ignore css and txt file
1843568 ignore file format
36315d2 first commit by us employee
36ea66d Latest commit from mumbai ec2-linux
895a6b3 first commit mumbai
0e106e4 adding first file from mumbai
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git branch
cat > branch1file
ls
git add .
git commit -m "branch1 first commit"
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
* branch1
  master
[root@ip-172-31-15-53 mumbaigit]# cat > branch1file
This file created in branch1
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css abc.java a.txt branch1file kroy.html mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "branch1 first commit"
[branch1 7124871] branch1 first commit
 1 file changed, 1 insertion(+)
 create mode 100644 branch1file
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git log --oneline
```

```
[root@ip-172-31-15-53 mumbaigit]# git log --oneline
7124871 (HEAD -> branch1) branch1 first commit
9d17a62 (origin/master, master) ignore css and txt file
1843568 ignore file format
36315d2 first commit by us employee
36ea66d Latest commit from mumbai ec2-linux
895a6b3 first commit mumbai
0e106e4 adding first file from mumbai
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git branch
git checkout master
git branch
ls
#branch1file will not be present in master branch
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
* branch1
  master
[root@ip-172-31-15-53 mumbaigit]# git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  kroy.html  mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# █
```

```
ls
git branch
git checkout branch1
ls
cat > secondbranchfile
ls
git checkout master
ls
#secondbranchfile will be visible in master also because this file is
in workspace not in staging or committed area
```

```
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  kroy.html  mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# git branch
  branch1
* master
[root@ip-172-31-15-53 mumbaigit]# git checkout branch1
Switched to branch 'branch1'
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  branch1file  kroy.html  mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# cat secondbranchfile
cat: secondbranchfile: No such file or directory
[root@ip-172-31-15-53 mumbaigit]# cat > secondbranchfile
This is second file from branch1
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  branch1file  kroy.html  mumbaifile1  secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
[root@ip-172-31-15-53 mumbaigit]# git branch
  branch1
* master
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  kroy.html  mumbaifile1  secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git branch
ls
git checkout branch1
ls
git add .
git commit -m "second commit from branch1"
git checkout master
ls
#now secondbranchfile will not be visible in master branch
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
  branch1
* master
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  kroy.html  mumbaifile1  secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# git checkout branch1
Switched to branch 'branch1'
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  branch1file  kroy.html  mumbaifile1  secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "second commit from branch1"
[branch1 9c971e9] second commit from branch1
1 file changed, 1 insertion(+)
create mode 100644 secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  kroy.html  mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# 
```

MERGE

```
git branch
ls
git merge branch1
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
  branch1
* master
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  kroy.html  mumbaifile1
[root@ip-172-31-15-53 mumbaigit]# git merge branch1
Updating 9d17a62..9c971e9
Fast-forward
 branch1file  | 1 +
 secondbranchfile | 1 +
 2 files changed, 2 insertions(+)
 create mode 100644 branch1file
 create mode 100644 secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  abc.java  a.txt  branch1file  kroy.html  mumbaifile1  secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git log --oneline
```

```
[root@ip-172-31-15-53 mumbaigit]# git log --oneline
9c971e9 (HEAD -> master, branch1) second commit from branch1
7124871 branch1 first commit
9d17a62 (origin/master) ignore css and txt file
1843568 ignore file format
36315d2 first commit by us employee
36ea66d Latest commit from mumbai ec2-linux
895a6b3 first commit mumbai
0e106e4 adding first file from mumbai
[root@ip-172-31-15-53 mumbaigit]# 
```

conflict

```
git branch
#master
ls
cat > newfile
git add .
git commit -m "added newfile"
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
branch1
* master
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css abc.java a.txt branch1file kroy.html mumbaifile1 secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# cat > newfile
hello kishan
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "added newfile"
[master da7608f] added newfile
1 file changed, 1 insertion(+)
create mode 100644 newfile
```

```
git chekout branch1
cat > newfile
git add .
git commit -m "added newfile from branch1 "
```

```
[root@ip-172-31-15-53 mumbaigit]# git checkout branch1
Switched to branch 'branch1'
Your branch is up to date with 'origin/branch1'.
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css abc.java a.txt branch1file kroy.html mumbaifile1 secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# cat > newfile
hi kishan
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "added newfile from branch1"
[branch1 703cffc] added newfile from branch1
1 file changed, 1 insertion(+)
create mode 100644 newfile
```

```
git checkout master
git merge branch1
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
* branch1
  master
[root@ip-172-31-15-53 mumbaigit]# git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)
[root@ip-172-31-15-53 mumbaigit]# git merge branch1
CONFLICT (add/add): Merge conflict in newfile
Auto-merging newfile
Automatic merge failed; fix conflicts and then commit the result.
[root@ip-172-31-15-53 mumbaigit]# 
```

```
cat newfile
vi newfile
git add .
git commit -m "resolve conflict"
git log --oneline
```

```
[root@ip-172-31-15-53 mumbaigit]# cat newfile
<<<<< HEAD
hello kishan
=====
hi kishan
>>>>> branch1
[root@ip-172-31-15-53 mumbaigit]# vi newfile
[root@ip-172-31-15-53 mumbaigit]# cat newfile

hello kishan
hi kishan
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "resolve conflict"
[master de96548] resolve conflict
[root@ip-172-31-15-53 mumbaigit]# git log --oneline
de96548 (HEAD -> master) resolve conflict
703cffc (branch1) added newfile from branch1
da7608f added newfile
9c971e9 (origin/master, origin/branch1) second commit from branch1
7124871 branch1 first commit
9d17a62 ignore css and txt file
1843568 ignore file format
```

Stashing

```
git branch
ls
touch demofile
git add .
git commit -m "demo file added"
```

```
[root@ip-172-31-15-53 mumbaigit]# git branch
  branch1
* master
[root@ip-172-31-15-53 mumbaigit]# ls
abc.css  a.txt      kroy.html  newfile
abc.java  branch1file  mumbaifile1  secondbranchfile
[root@ip-172-31-15-53 mumbaigit]# touch demofile
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "demo file added"
[master aa20620] demo file added
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 demofile
[root@ip-172-31-15-53 mumbaigit]# 
```

```
vi demofile
cat demofile
```

```
[root@ip-172-31-15-53 mumbaigit]# vi demofile
[root@ip-172-31-15-53 mumbaigit]# cat demofile
first code
[root@ip-172-31-15-53 mumbaigit]# █
```

```
git stash
cat demofile
#data of demofile gone to stash
```

```
[root@ip-172-31-15-53 mumbaigit]# cat demofile
first code
[root@ip-172-31-15-53 mumbaigit]# git stash
Saved working directory and index state WIP on master: aa20620 demo file added
[root@ip-172-31-15-53 mumbaigit]# cat demofile
[root@ip-172-31-15-53 mumbaigit]# █
```

```
git stash list
vi demofile
cat demofile
git stash
git stash list
cat demofile
```

```
[root@ip-172-31-15-53 mumbaigit]# git stash list
stash@{0}: WIP on master: aa20620 demo file added
[root@ip-172-31-15-53 mumbaigit]# vi demofile
[root@ip-172-31-15-53 mumbaigit]# cat demofile
second code for client
[root@ip-172-31-15-53 mumbaigit]# git stash
Saved working directory and index state WIP on master: aa20620 demo file added
[root@ip-172-31-15-53 mumbaigit]# git stash list
stash@{0}: WIP on master: aa20620 demo file added
stash@{1}: WIP on master: aa20620 demo file added
[root@ip-172-31-15-53 mumbaigit]# cat demofile
[root@ip-172-31-15-53 mumbaigit]# █
```

```
git stash apply stash@{1}
```

```
[root@ip-172-31-15-53 mumbaigit]# cat demofile
[root@ip-172-31-15-53 mumbaigit]# git stash apply stash@{1}
On branch master
Your branch is ahead of 'origin/master' by 4 commits.
  (use "git push" to publish your local commits)

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:   demofile

no changes added to commit (use "git add" and/or "git commit -a")
[root@ip-172-31-15-53 mumbaigit]# cat demofile
first code
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git add .
git commit -m "first code done"
```

```
[root@ip-172-31-15-53 mumbaigit]# cat demofile
first code
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "first code done"
[master ec8b2a8] first code done
 1 file changed, 1 insertion(+)
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git stash apply stash@{0}
#Merge conflict as file name is same and file is already committed
```

```
[root@ip-172-31-15-53 mumbaigit]# git stash apply stash@{0}
Auto-merging demofile
CONFLICT (content): Merge conflict in demofile
[root@ip-172-31-15-53 mumbaigit]# 
```

```
cat demofile
vi demofile
cat demofile
```

```
[root@ip-172-31-15-53 mumbaigit]# cat demofile
<<<<< Updated upstream
first code
=====
second code for client
>>>>> Stashed changes
[root@ip-172-31-15-53 mumbaigit]# vi demofile
[root@ip-172-31-15-53 mumbaigit]# cat demofile

first code

second code for client
[root@ip-172-31-15-53 mumbaigit]# 
```

```
git add .
git commit -m "stash conflict resolved"
```

```
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "stash conflict resolved"
[master bb1424d] stash conflict resolved
 1 file changed, 3 insertions(+)
[root@ip-172-31-15-53 mumbaigit]# 
```

GITRESET

```
cat > testfile
git add .
git status
git reset .
git status
#file will be removed from staging area
```

```
[root@ip-172-31-15-53 mumbaigit]# cat > testfile
test data for git reset
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# 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:  testfile

[root@ip-172-31-15-53 mumbaigit]# git reset .
[root@ip-172-31-15-53 mumbaigit]# 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)
    testfile

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-15-53 mumbaigit]# 
```

GITREVERT

```
cat > revertfile
git add .
git commit -m "this is revertfilecommit"
cat > revertfile
git add .
git commit -m "wrong code added"
```

```
[root@ip-172-31-15-53 mumbaigit]# cat > revertfile
revertcode
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "this is revertfilecommit"
[master 91738a7] this is revertfilecommit
1 file changed, 1 insertion(+)
 create mode 100644 revertfile
[root@ip-172-31-15-53 mumbaigit]# cat > revertfile
wrong code
[root@ip-172-31-15-53 mumbaigit]# git add .
[root@ip-172-31-15-53 mumbaigit]# git commit -m "wrong code added"
[master 1c07eed] wrong code added
1 file changed, 1 insertion(+), 1 deletion(-)
[root@ip-172-31-15-53 mumbaigit]# 
```

```
cat revertfile
git log
```

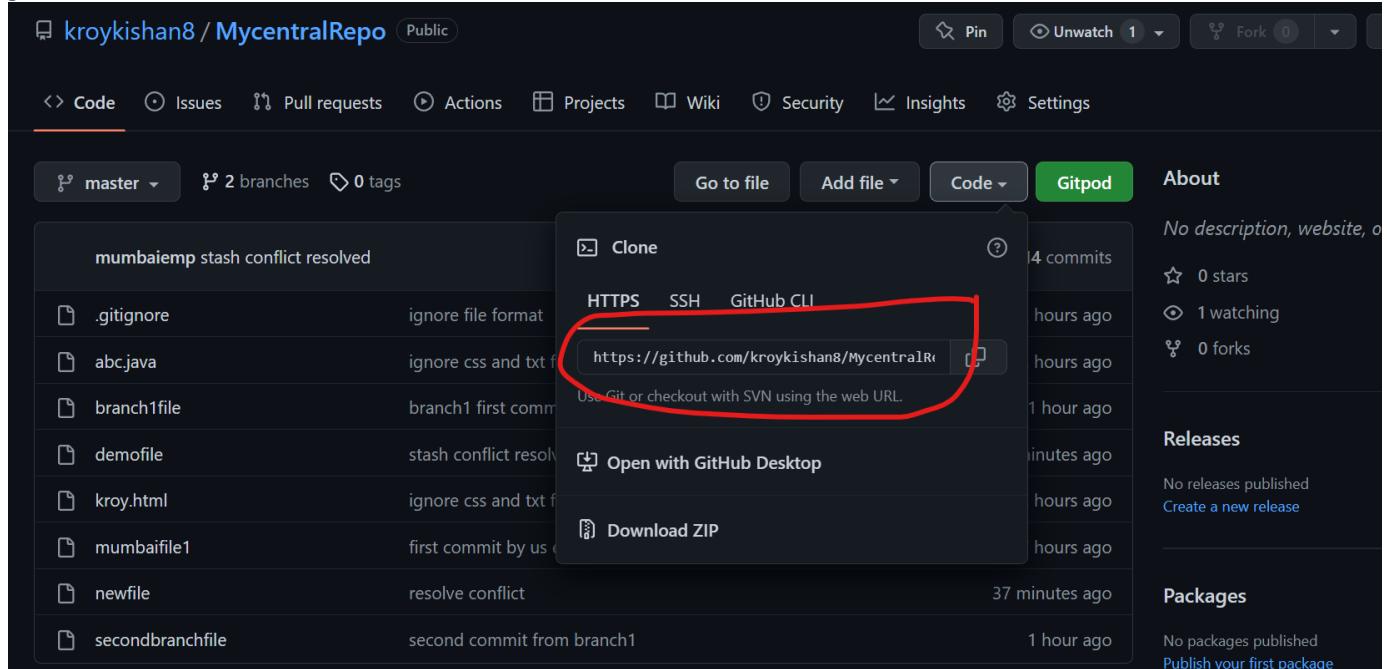
```
[root@ip-172-31-15-53 mumbaigit]# cat revertfile
wrong code
[root@ip-172-31-15-53 mumbaigit]# git log
commit 1c07eedea0720258c94952e96c69e8e61be74a9f (HEAD -> master)
Author: mumbaiemp <mumbaiemp@gmail.com>
Date:   Fri Jul 8 16:15:08 2022 +0000

    wrong code added
```

```
git revert <commit id>
```

```
[root@ip-172-31-15-53 mumbaigit]# git revert 1c07eedea0720258c94952e96c69e8e61be74a9f
[master 2fdeb9e] Revert "wrong code added "please ignore this commit
 1 file changed, 1 insertion(+), 1 deletion(-)
[root@ip-172-31-15-53 mumbaigit]# cat revertfile
revertcode
[root@ip-172-31-15-53 mumbaigit]#
```

gitclone



The screenshot shows a GitHub repository page for 'kroykishan8 / MycentralRepo'. The 'Code' tab is selected. The 'Clone' menu is open, showing three options: 'HTTPS', 'SSH', and 'GitHub CLI'. The 'HTTPS' option is highlighted with a red box. Below the menu, a note says 'Use Git or checkout with SVN using the web URL.' The 'About' section on the right shows 14 commits, 0 stars, 1 watching, and 0 forks. The 'Releases' and 'Packages' sections are also visible.

```
git clone https://github.com/kroykishan8/MycentralRepo.git
```

```
[root@ip-172-31-15-53 ec2-user]# ls
mumbaigit
[root@ip-172-31-15-53 ec2-user]# git clone https://github.com/kroykishan8/MycentralRepo.git
Cloning into 'MycentralRepo'...
remote: Enumerating objects: 41, done.
remote: Counting objects: 100% (41/41), done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 41 (delta 11), reused 39 (delta 9), pack-reused 0
Receiving objects: 100% (41/41), done.
Resolving deltas: 100% (11/11), done.
[root@ip-172-31-15-53 ec2-user]# ls
mumbaigit  MycentralRepo
[root@ip-172-31-15-53 ec2-user]# 
[root@ip-172-31-15-53 ec2-user]# cd MycentralRepo/
[root@ip-172-31-15-53 MycentralRepo]# ls
abc.java      demofile    mumbaifile1  secondbranchfile
branch1file  kroy.html  newfile
[root@ip-172-31-15-53 MycentralRepo]# 
```

finished.....