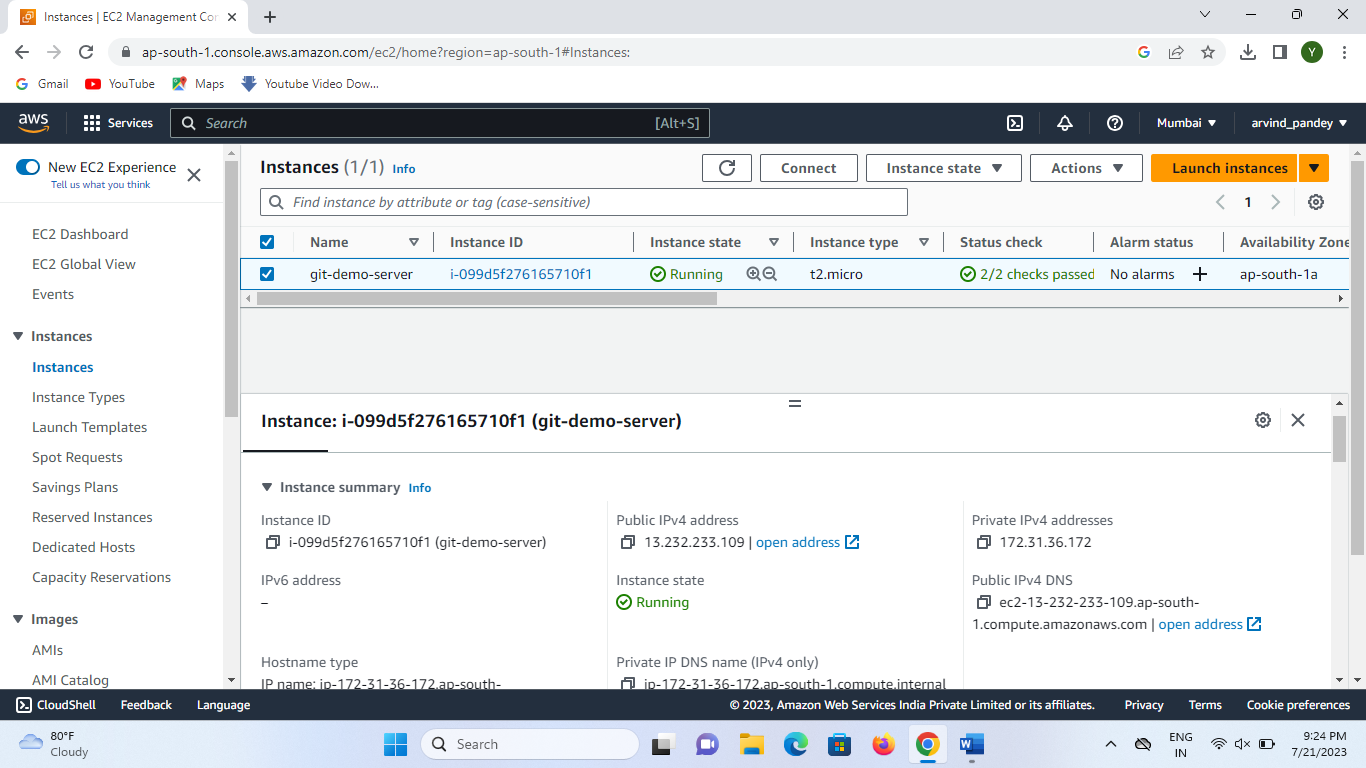
# **Step 1 ;- Launch a Ubuntu machine with EC2 Instance**

## Using Ubuntu AMI with t2.micro instance type

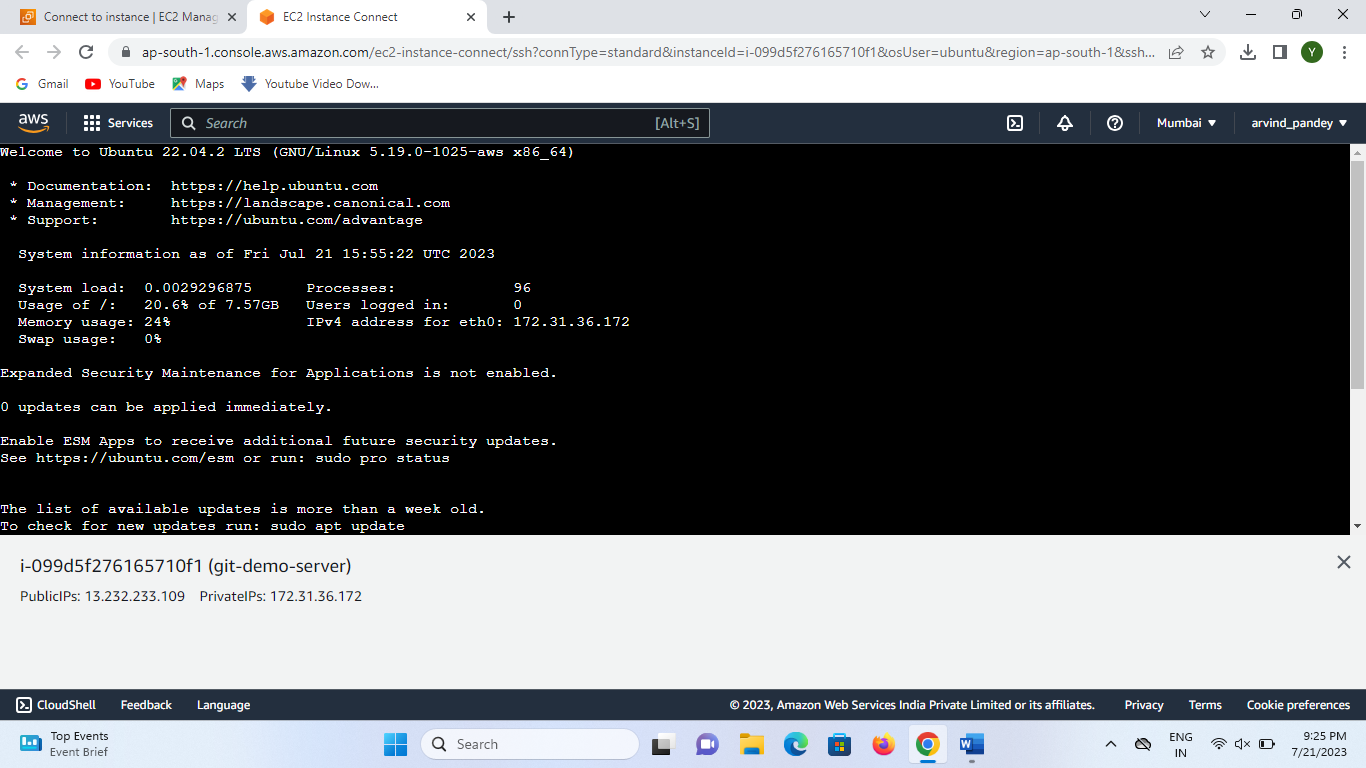
## Create key pair to securely connect to your instance

## Create A security group set firewall rules that control the traffic for your instance



# **Step 2;- Select the EC2 instance and click on Connect button**

## Ubuntu terminals are ready to use



# **Step 4;- Perform common git commands**

## Create Program workspace(project directory) and add file in this.

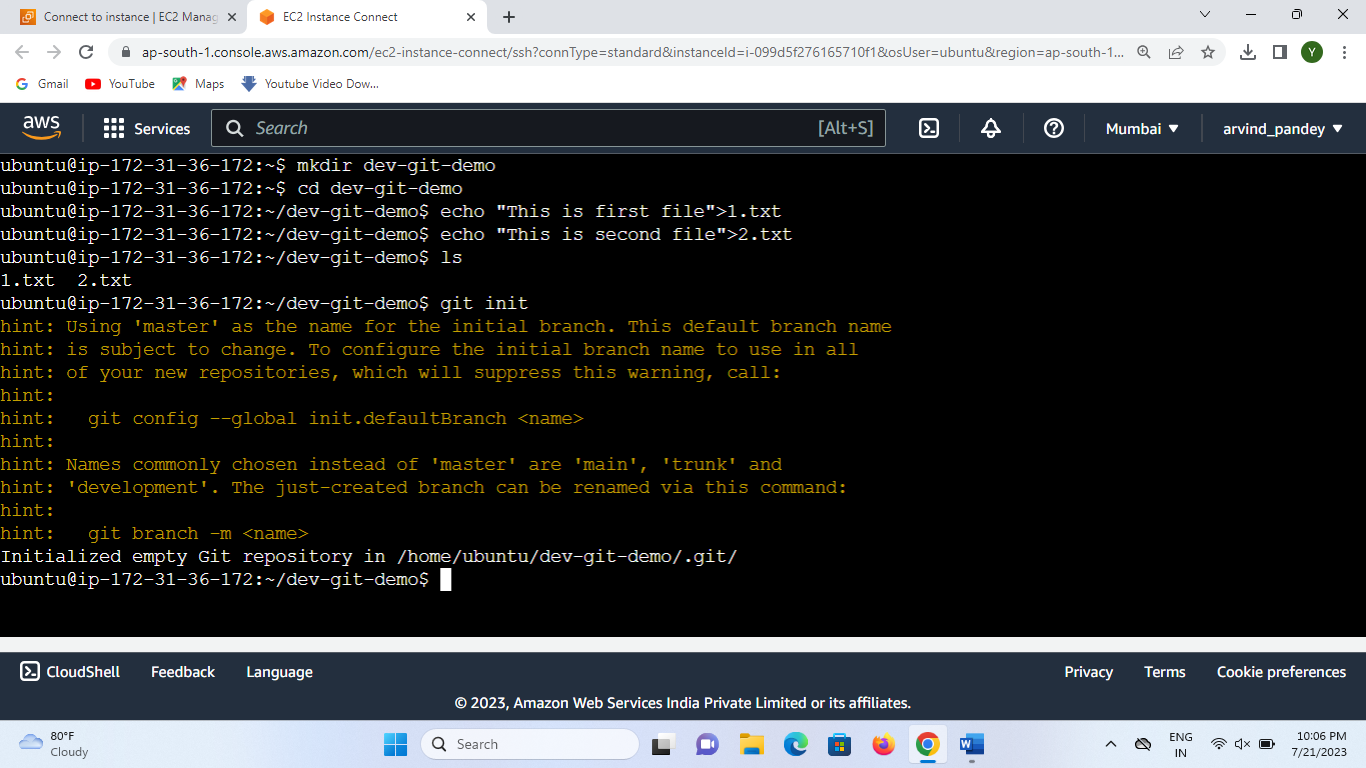
## Echo is the command which is use to display the argument we passed.

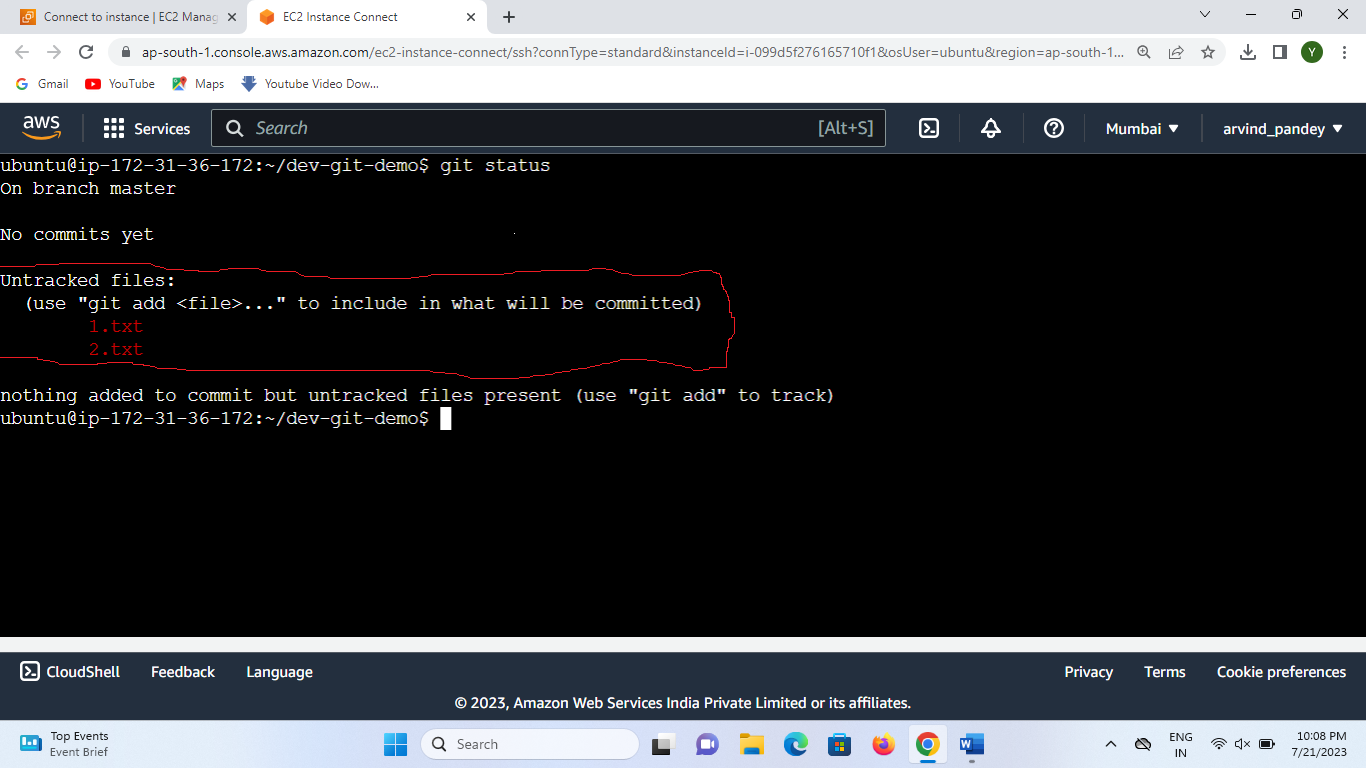
## git init use to convert project directory into working directory.

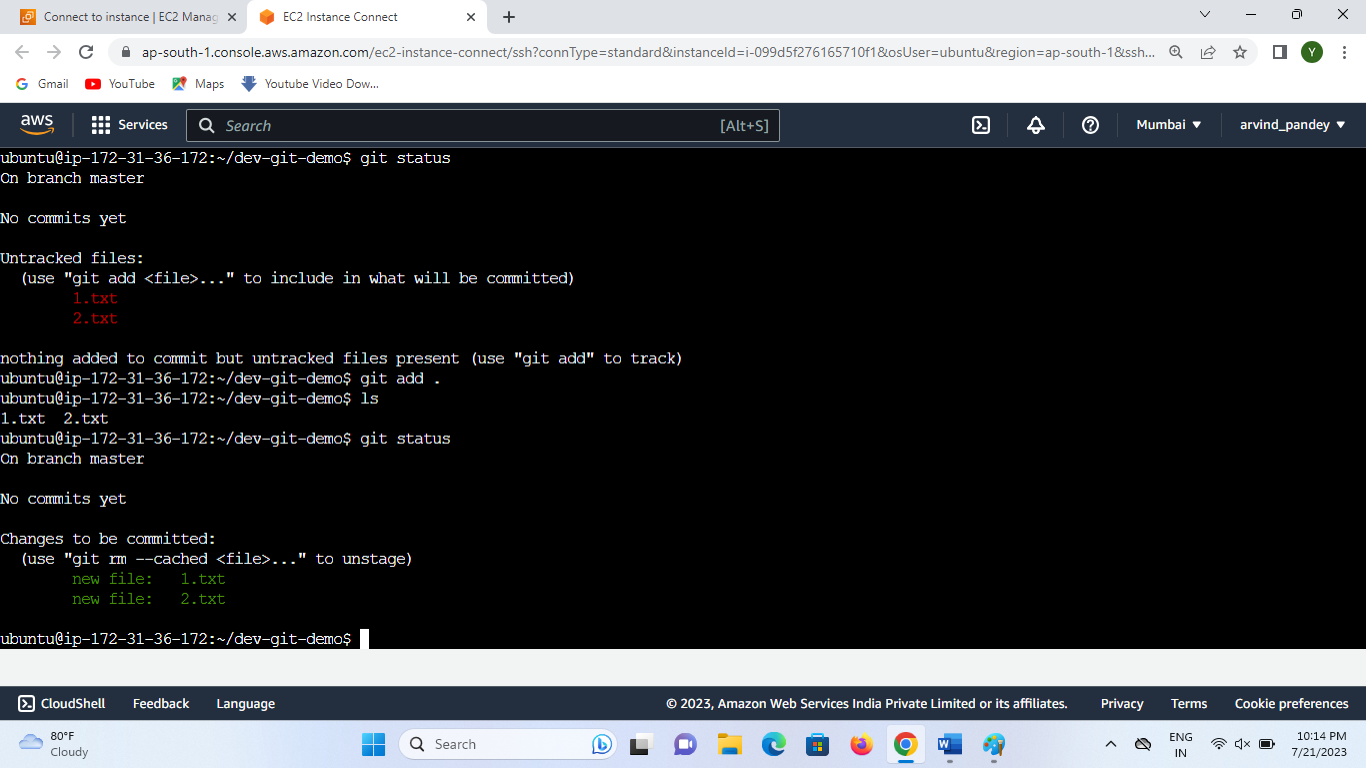
## git init created a local repository.

## Add our files into staging area (i.e 1.txt & 2.txt)

## git add (1.txt 2.txt)to add file in local repository



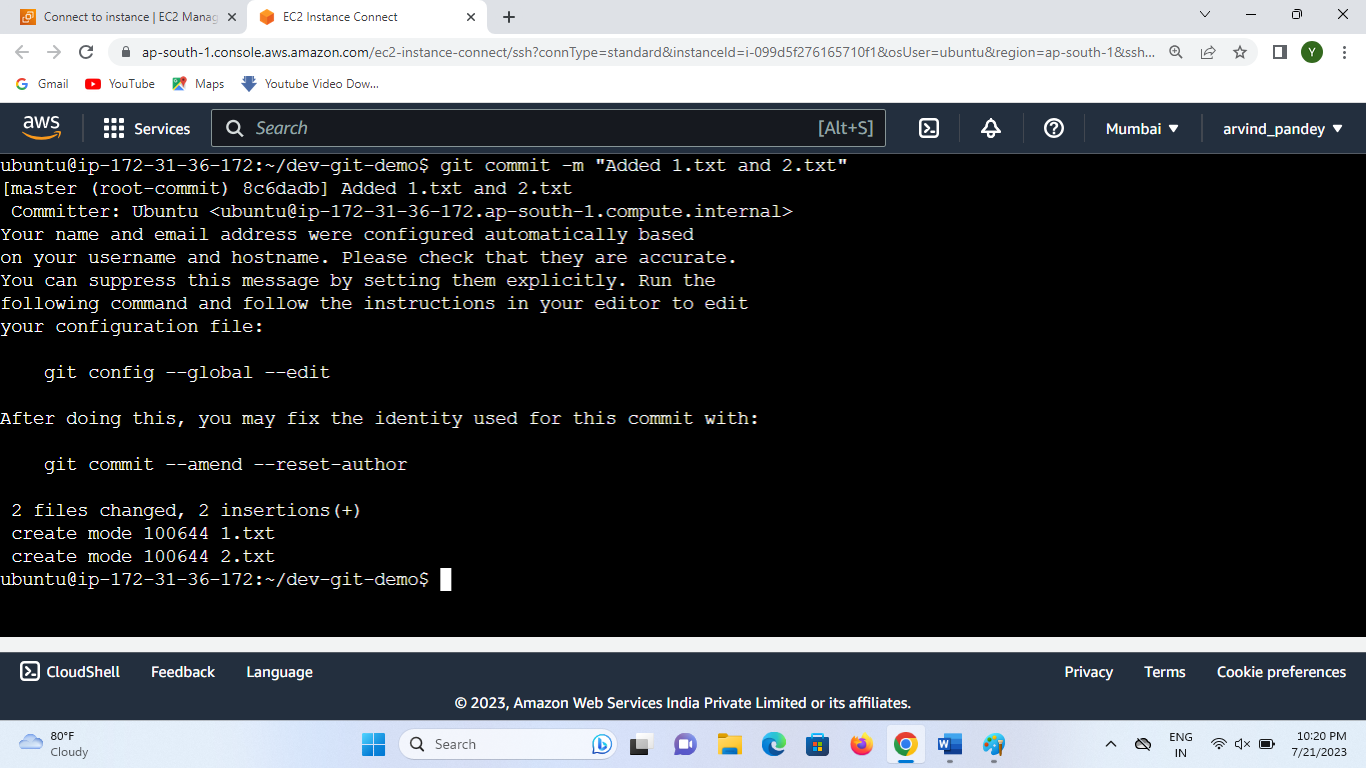




# **Step 5;- Move Staging Area to Commit**

## git commit save the file in local repository

## It’s a temporary storage area (unstage)



# **Step 6;- git log**

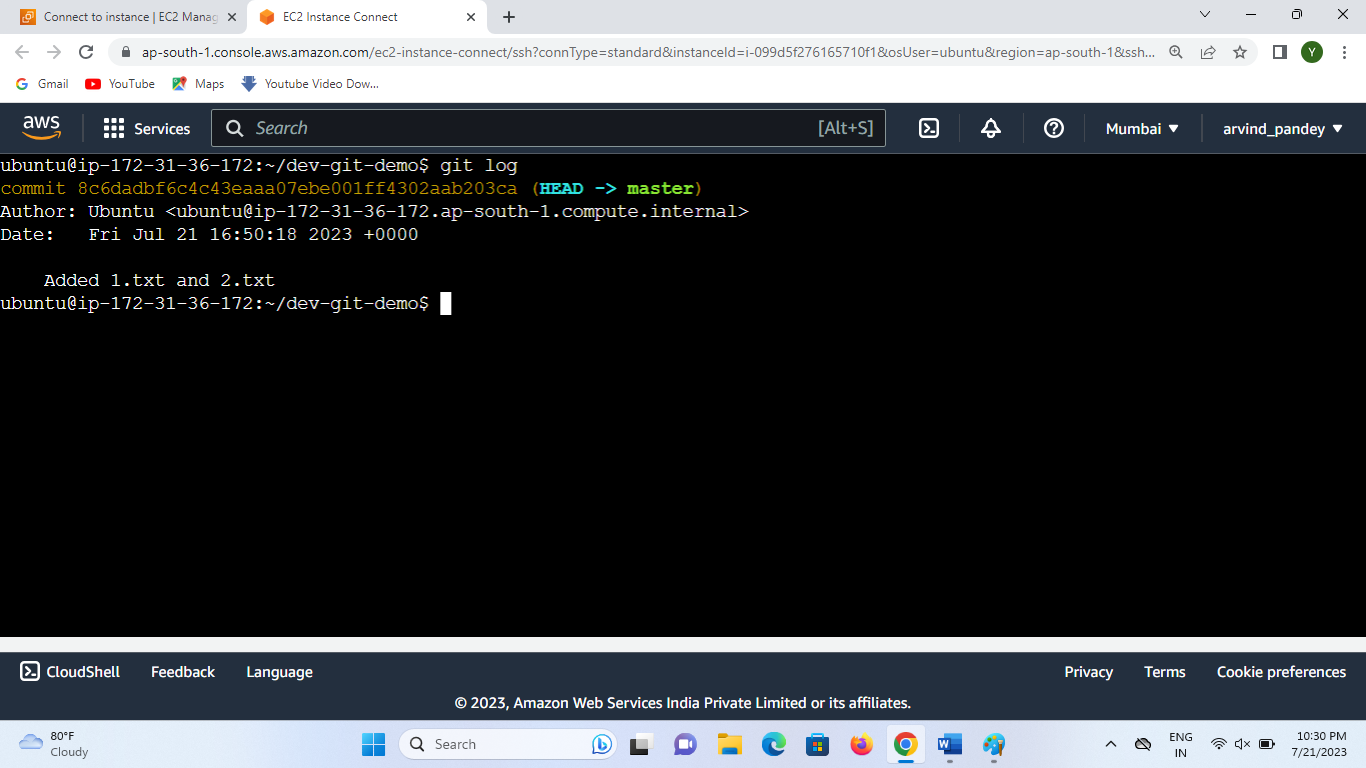
## git log will give entire content of commit

## msg of an Author (who did this commit) and commit ID

## date and time along with commit msg or what user has done

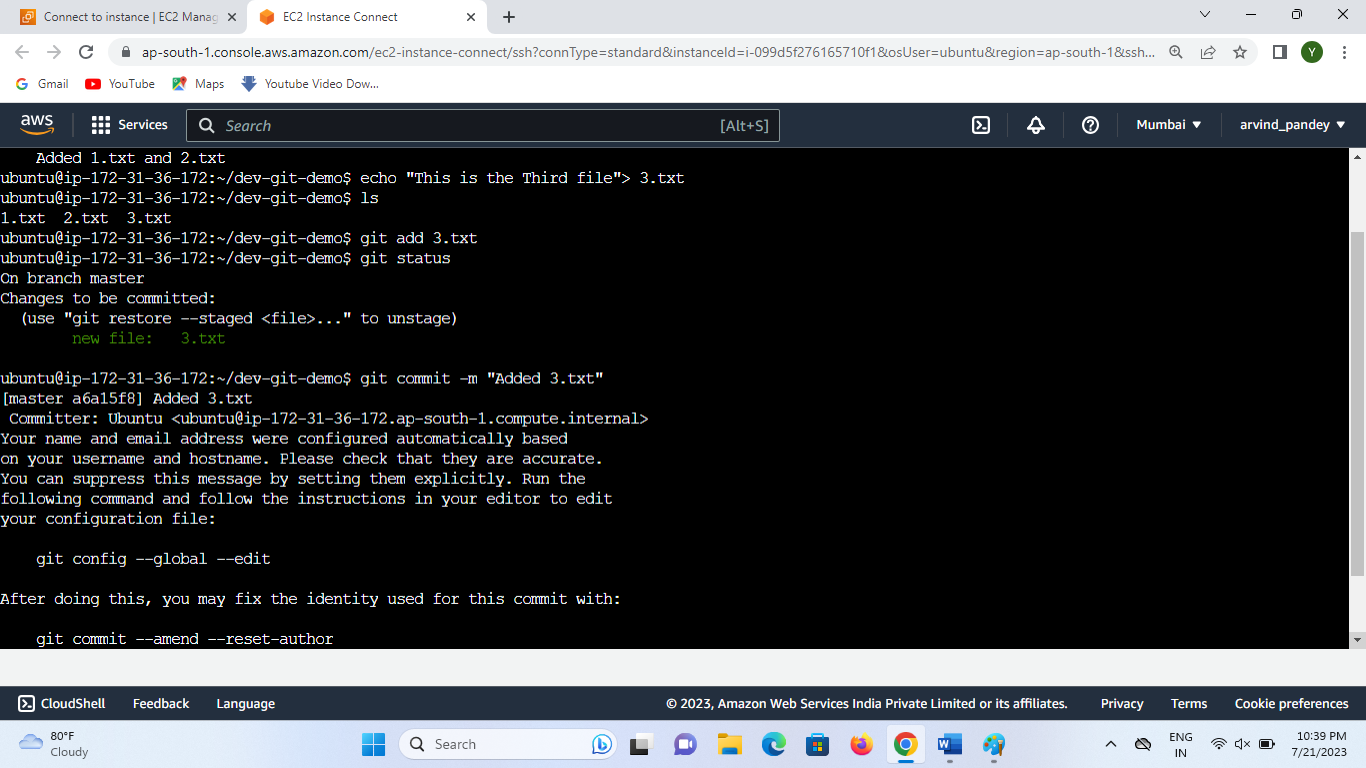
## master is the Branch that we are using (default branch is Master)

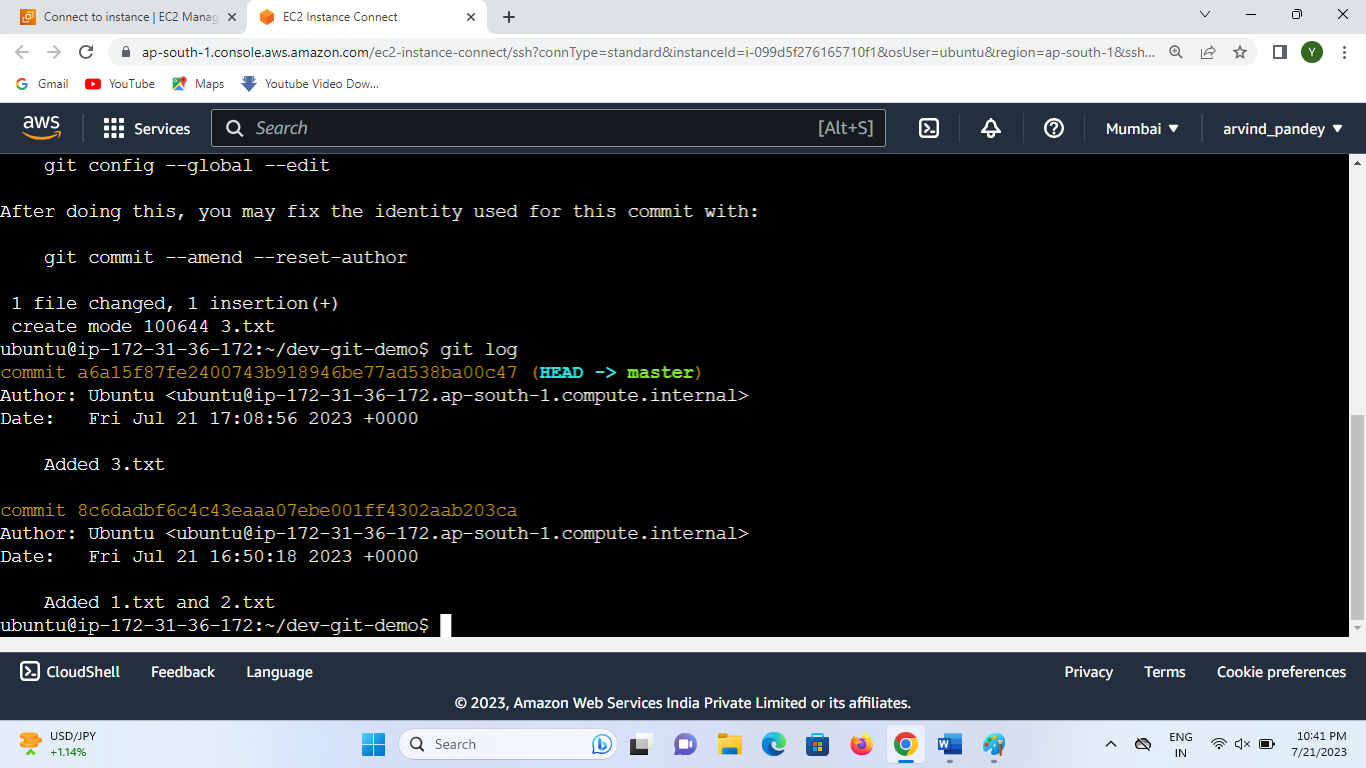
## pointer to the latest commit is called as Head (it will refer to the latest commit)



# Step 7;-

## Add a Third file



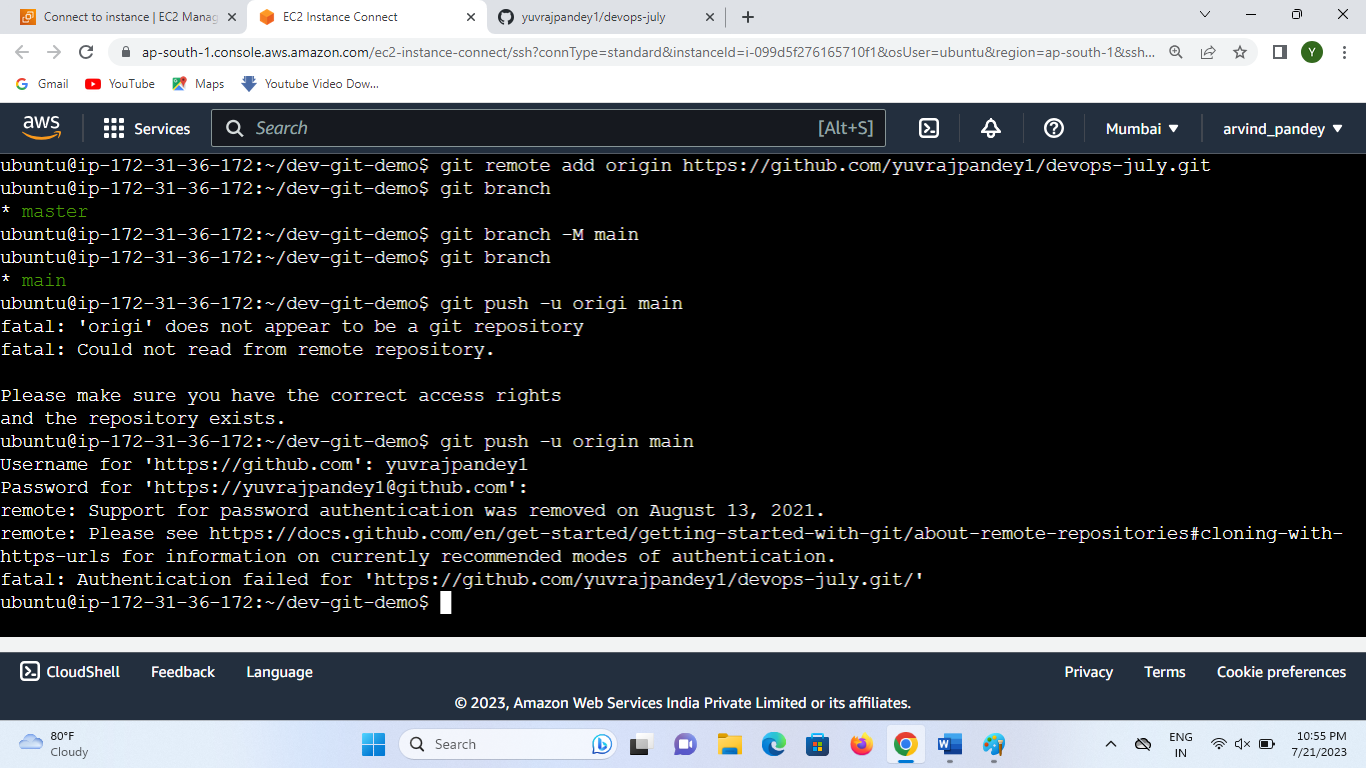


# **Step 8 ;- Push all changes into GitHub account**

## Create a repository in GitHub

## Copy remote repository command

## Change branch name master to main



# **Step 9;- Generate PAT (Personal Access Token) to overcome this error**

## Steps to create Personal Access Token

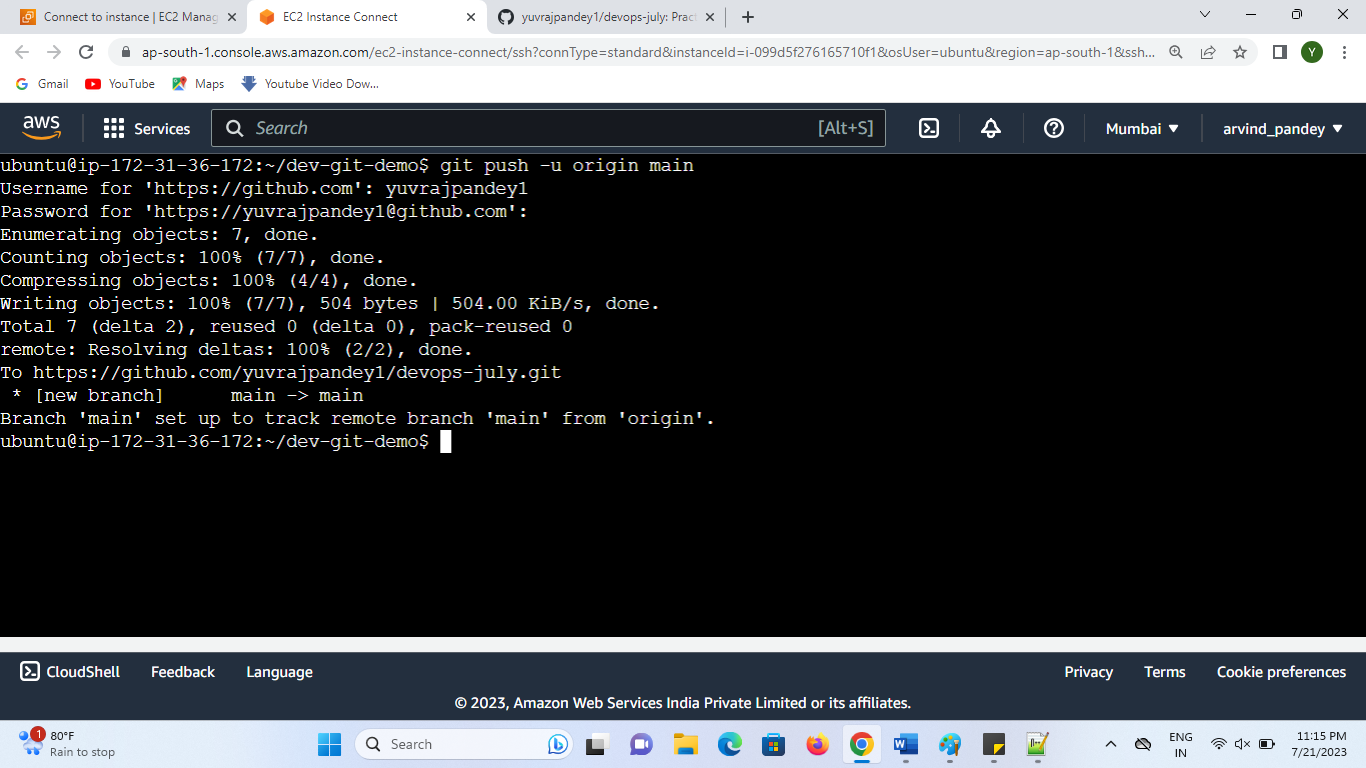
### Go to the GitHub profile to settings

### From settings go to Developer settings

### <https://github.com/settings/apps>

### Personal access tokens ->Tokens(classic)

### Generate new token



## **Step 10;- What are the need of the Branch**

## If you want to add new file we should not directly push to Master Branch

## Create a new Branch under Master Branch and then push new file

## Master/Main branch will contain only the finalized code

## Only Main code will be their in Master/Main Branch

### If any error or defect is their in Branch it will Not effect the Master/Main Branch

### After doing changes at last we can merge in Master/Main Branch

### Created new branch feature-4 (git branch feature-4)

