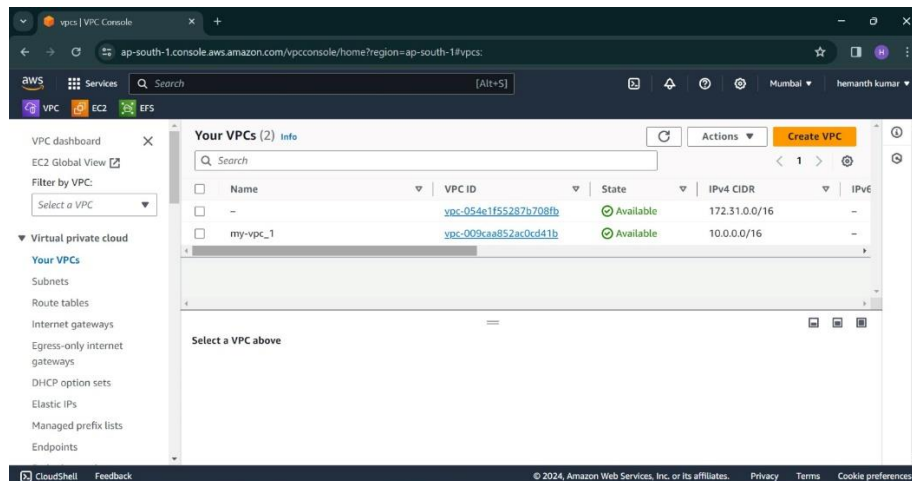


ASSIGNMENT

Create a VPC with 2 subnets and 2 route tables and internet gateway

Create a VPC(virtual private cloud)

Now click on Create VPC to create our custom VPC



Now We have to give the details for our VPC and

finally click on Create VPC

Subnet 1 of 1

Subnet name

Create a tag with a key of 'Name' and a value that you specify.

public_subnet

The name can be up to 256 characters long.

Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

Asia Pacific (Mumbai) / ap-south-1a

IPv4 VPC CIDR block [Info](#)

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

10.0.0.0/16

IPv4 subnet CIDR block

10.0.1.0/24

256 IPs

▼ Tags - optional

Key Value - optional

Now we created our custom VPC successfully.

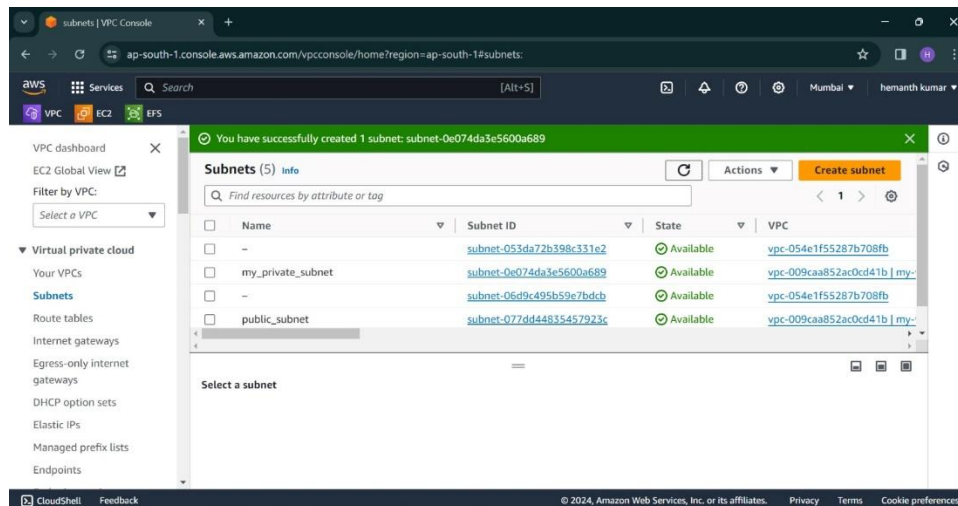
Now click on Subnets to create Subnets to our custom VPC

Then create two subnets

public and private

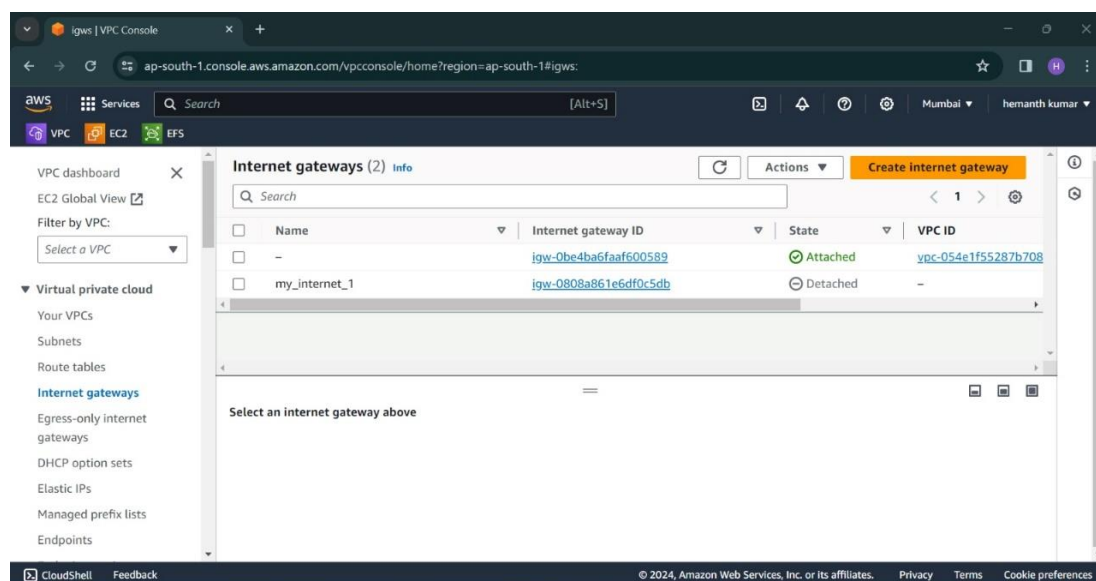
We have select the our custom VPC-ID, Subnet name, choose only one availability Zone, IPv4 subnet CIDR block to give the IP, then finally create subnet

Now we created two Subnets to our custom VPC successfully

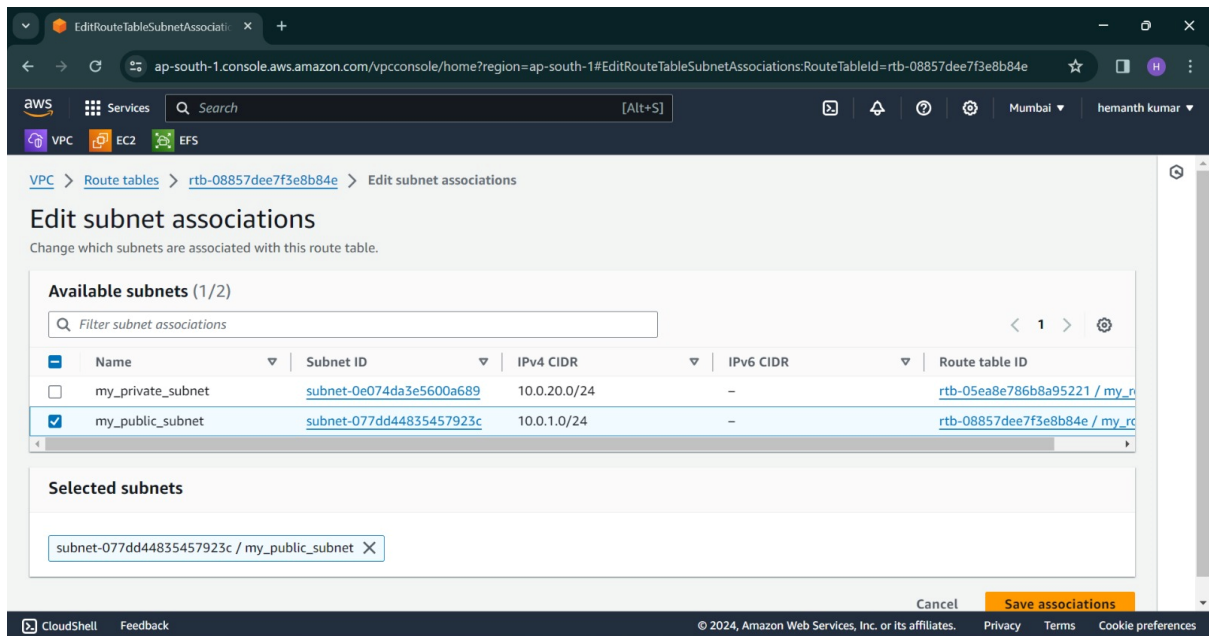


Now, we have give name to our internet gateway and finally click on Create internet gateway.

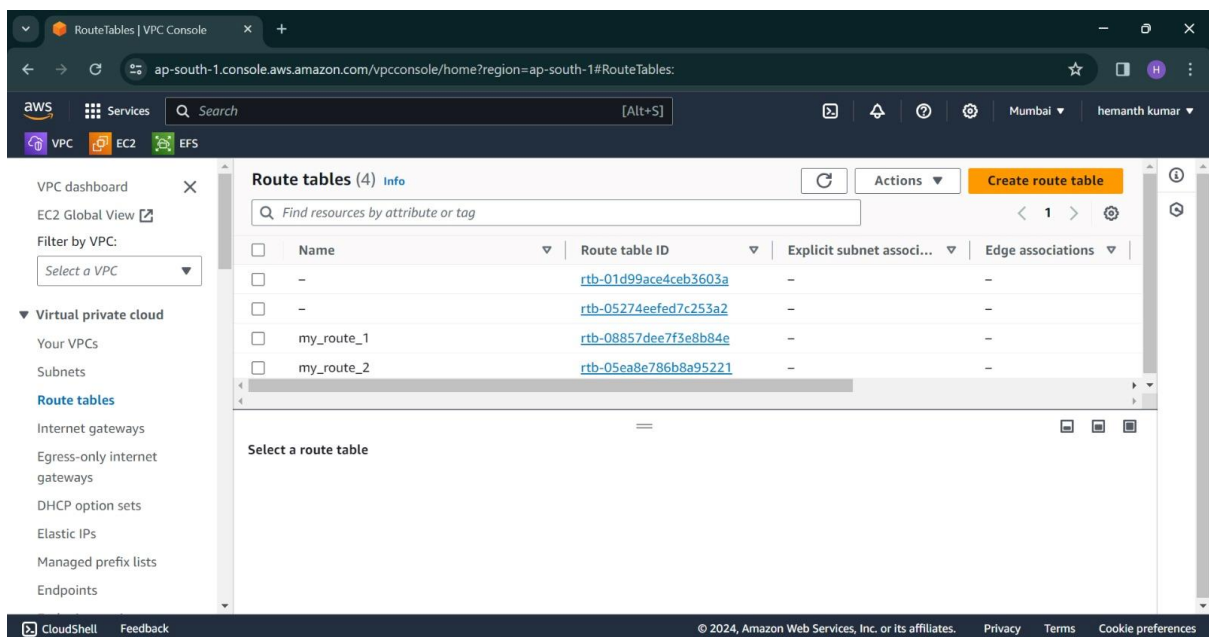
Public subnet:



Private subnet:



Now we have to create 2 route tables (one is public and another one is private). Click on Route tables from menu bar and click on Create route table



click on Add route. Select 0.0.0.0/0 as Destination

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
<input type="text" value="10.0.0.0/16"/>	<input type="text" value="local"/>		

finally click on Save changes

Then click on Subnet associations and Edit subnet associations

Create one more route table (private-subnet) and associate with private subnet.

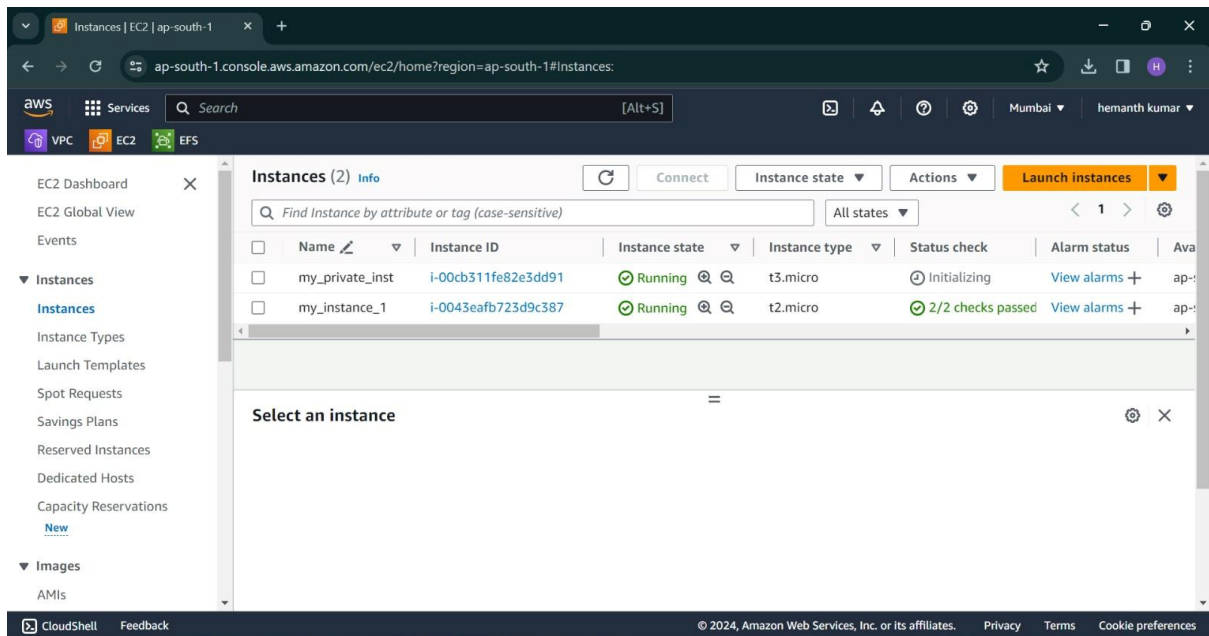
Now we created two route table

VPC with 2 subnets and 2 route tables and internet gateway successfully created.

Create one EC2 Instances

Search for EC2 in search space of AWS home page and click on Ec2

Now Create one ec2 instance



Then launch the ec2 instance, then We have to give the details for our ec2(EBS)Instance and then we have mention some details like we have name, OS type to start, instance type, keypair(login),network setting.

click on Launch instance

Now we created one EC2 Instance successfully.

```

root@ip-10-0-1-132:~
hemant@HeMu MINGW64 ~/Desktop
$ chmod 400 "onee.pem"

hemant@HeMu MINGW64 ~/Desktop
$ ssh -i "onee.pem" ec2-user@13.233.22.169
The authenticity of host '13.233.22.169 (13.233.22.169)' can't be established.
ED25519 key fingerprint is SHA256:8Hb0gf+3UdicPcoRYNkDU2IUQ5NYzkERGgiUHNPGHu1E.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.233.22.169' (ED25519) to the list of known hosts.

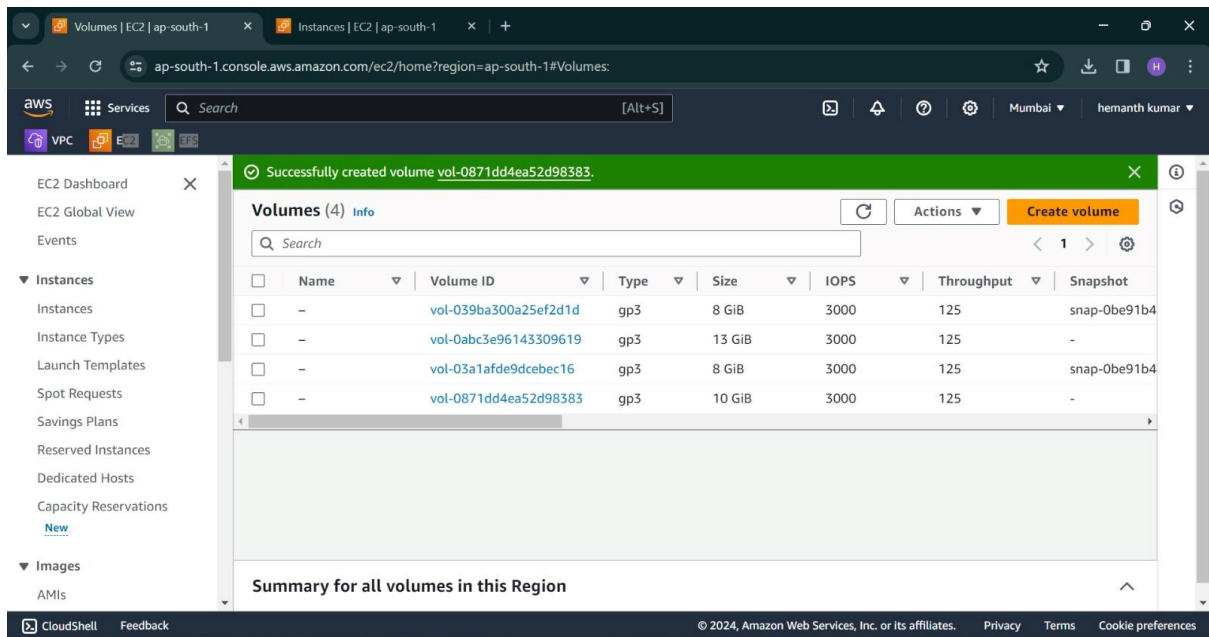
#_
#####          Amazon Linux 2023
~~\#####\
~~\###\
~~\#/\
~~\V~'~>
~~~
~~~.~.~
~~~/_/m/'~>

[ec2-user@ip-10-0-1-132 ~]$ sudo_i
-bash: sudo_i: command not found
[ec2-user@ip-10-0-1-132 ~]$ sudo -i
[root@ip-10-0-1-132 ~]#

```

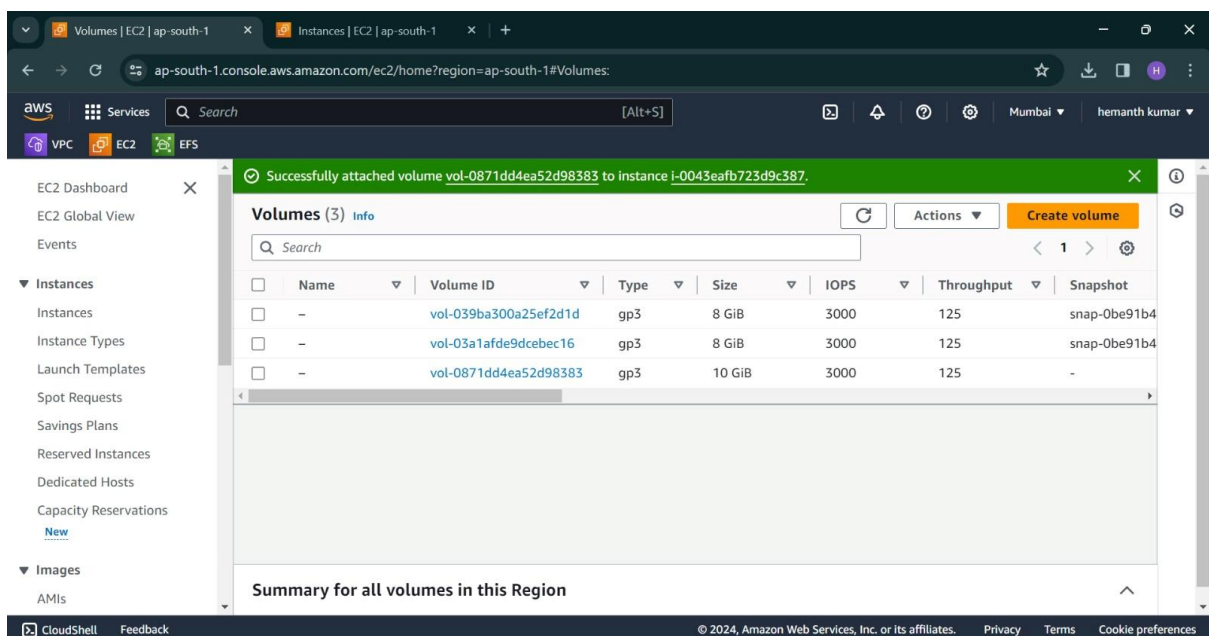
Now click on Elastic Block Store option from EC2 instance menu

Then click on volumes.



create a Volume in the volume setting so that We have to give the details for volume type, size, availability zone.

Then click on create volume



The volume created successfully

The volume is given 10 Gib.

Once the volume has been create click on actions in that click on attach volume

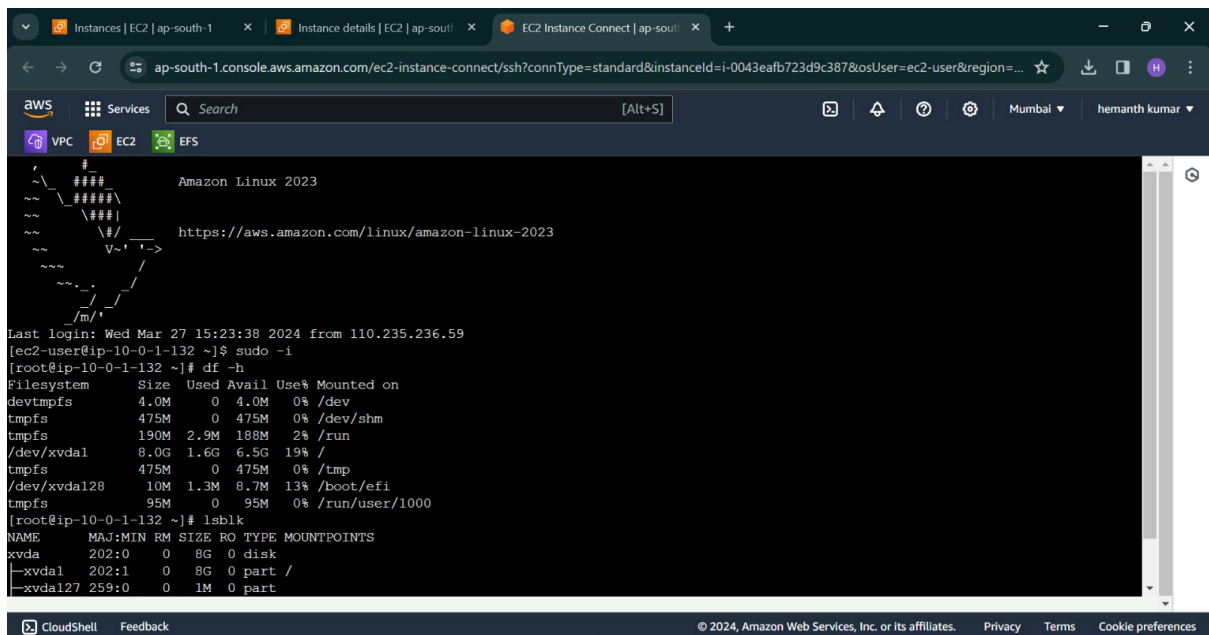
Click on SSH client in connect to instance page

Now go to Gitbash window

Run this commands, to give permission

```
chmod 400 "ebs.pem"
```

copy the ssh



```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

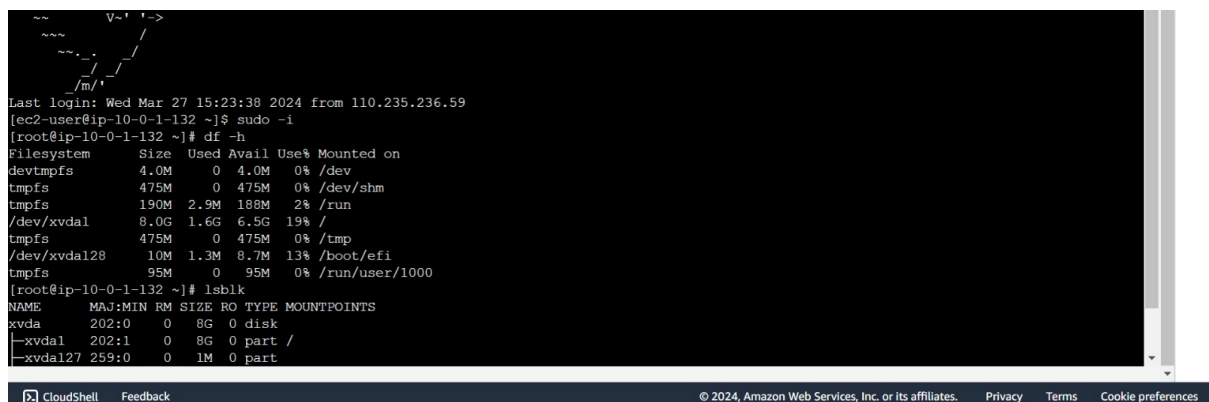
Last login: Wed Mar 27 15:23:38 2024 from 110.235.236.59
[ec2-user@ip-10-0-1-132 ~]$ sudo -i
[root@ip-10-0-1-132 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M   0  4.0M   0% /dev
tmpfs           475M   0  475M   0% /dev/shm
tmpfs           190M  2.9M  188M   2% /run
/dev/xvda1      8.0G  1.6G   6.5G  19% /
tmpfs           475M   0  475M   0% /tmp
/dev/xvda128    10M   1.3M   8.7M  13% /boot/efi
tmpfs           95M   0   95M   0% /run/user/1000
[root@ip-10-0-1-132 ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
xvda        202:0    0    8G  0 disk
└─xvda1     202:1    0    8G  0 part /
└─xvda127   259:0    0    1M  0 part
```

Run this commands

sudo -i (to change to root user)

df -h (to check the disk space)

lsblk (to list out block devices)



```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Wed Mar 27 15:23:38 2024 from 110.235.236.59
[ec2-user@ip-10-0-1-132 ~]$ sudo -i
[root@ip-10-0-1-132 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M   0  4.0M   0% /dev
tmpfs           475M   0  475M   0% /dev/shm
tmpfs           190M  2.9M  188M   2% /run
/dev/xvda1      8.0G  1.6G   6.5G  19% /
tmpfs           475M   0  475M   0% /tmp
/dev/xvda128    10M   1.3M   8.7M  13% /boot/efi
tmpfs           95M   0   95M   0% /run/user/1000
[root@ip-10-0-1-132 ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
xvda        202:0    0    8G  0 disk
└─xvda1     202:1    0    8G  0 part /
└─xvda127   259:0    0    1M  0 part
```

Then Run is command

Clear (to clear the screen)

Finally our volume is attached.