

IAC PROJECT(Domain:-Cloud Computing).

Automate MongoDB Backup.

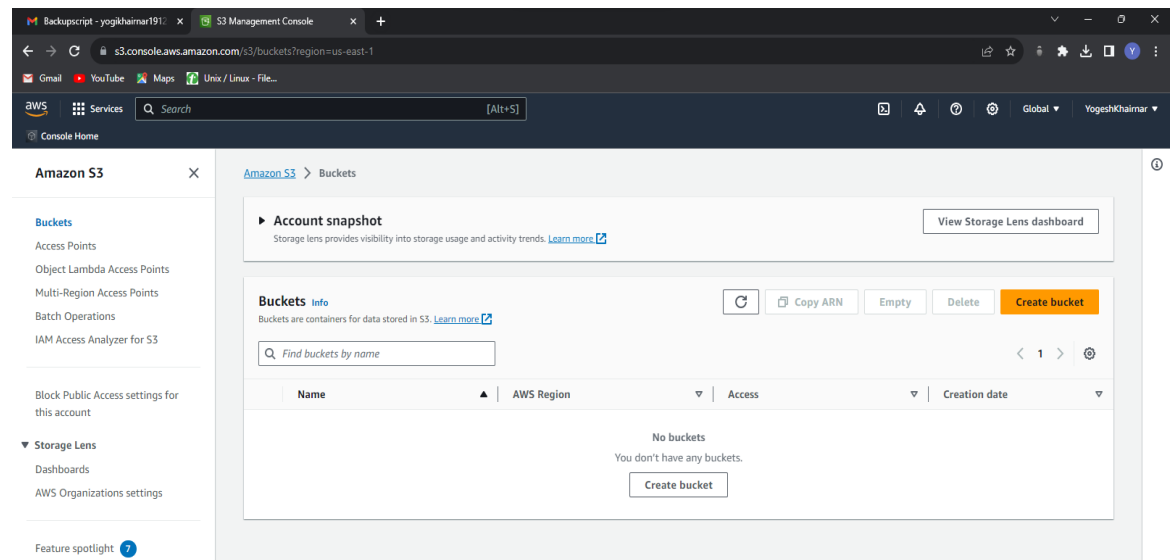
Requirements:-AWS Account,MongoDB.

AWS Services:-EC2,S3,IAM(Role).

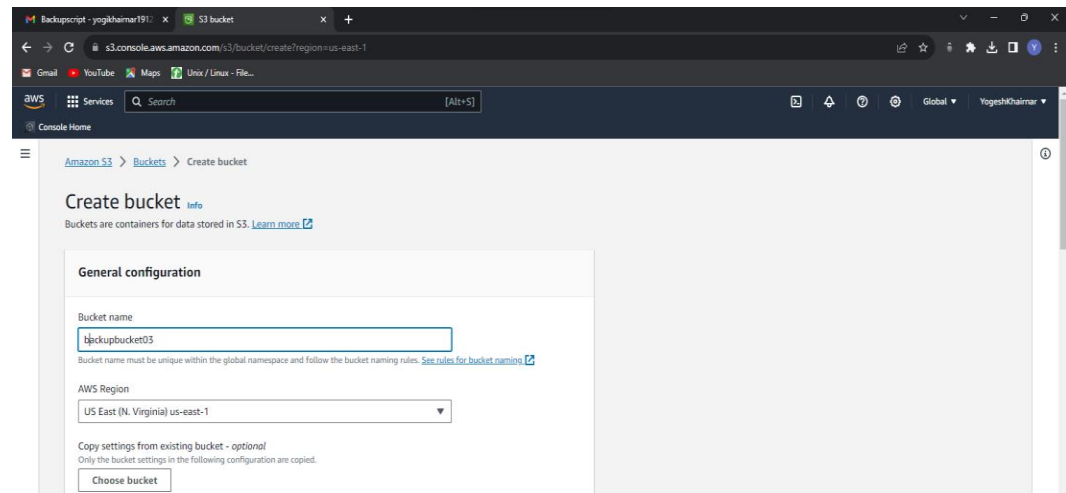
Steps:-

1. S3:-Create a bucket and inside that bucket create a backup folder.

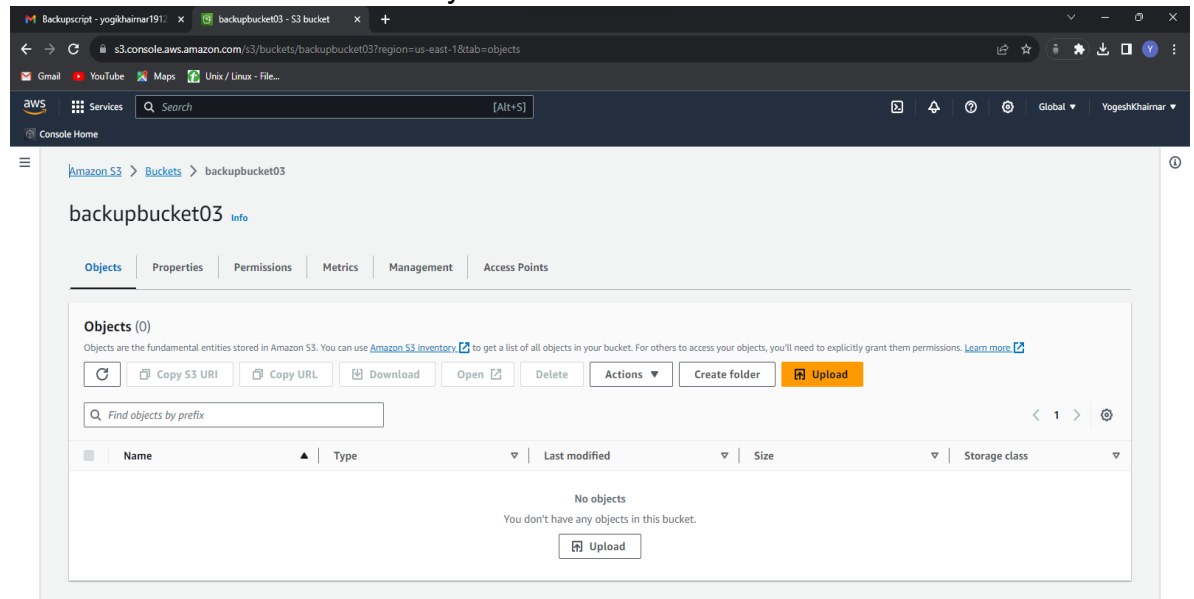
Login to aws account>Go to management console>search for s3>Bucket>Create bucket.



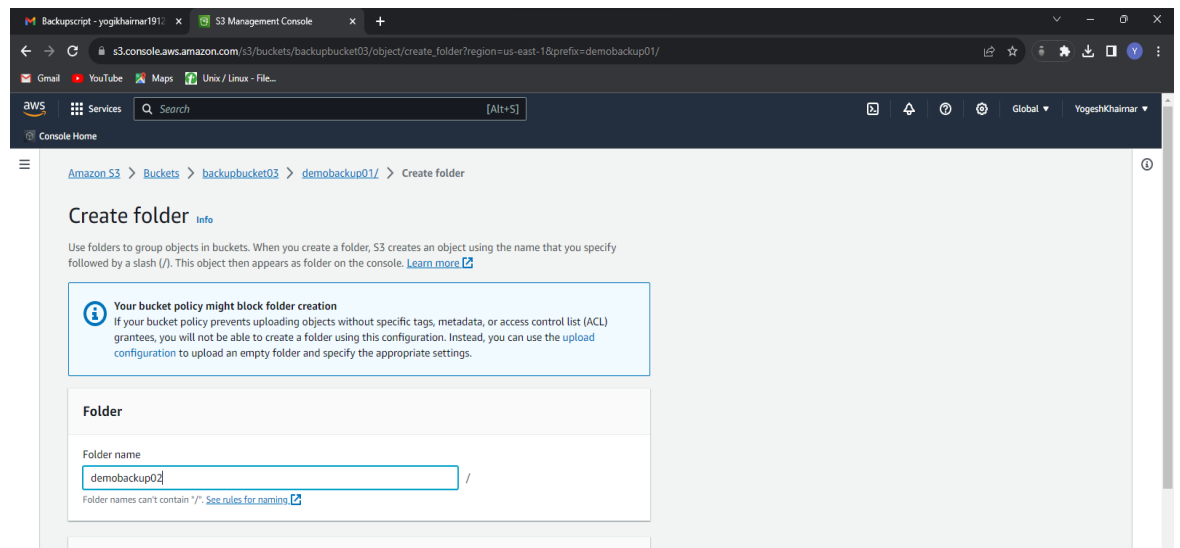
Give name of the bucket>Create bucket.



Now click on the bucket which you have created>Create Folder.

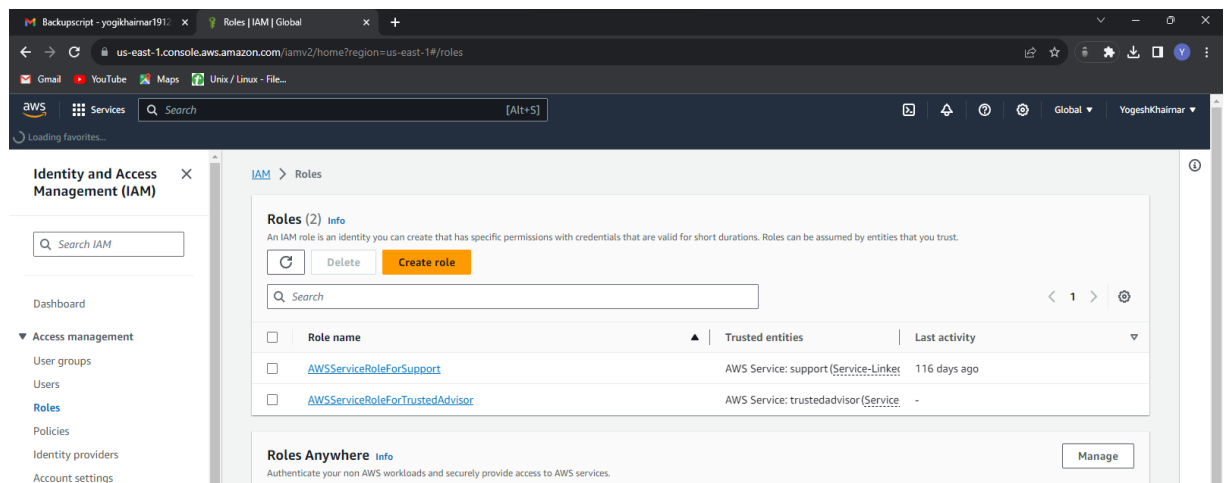


Give the name of folder>Create.

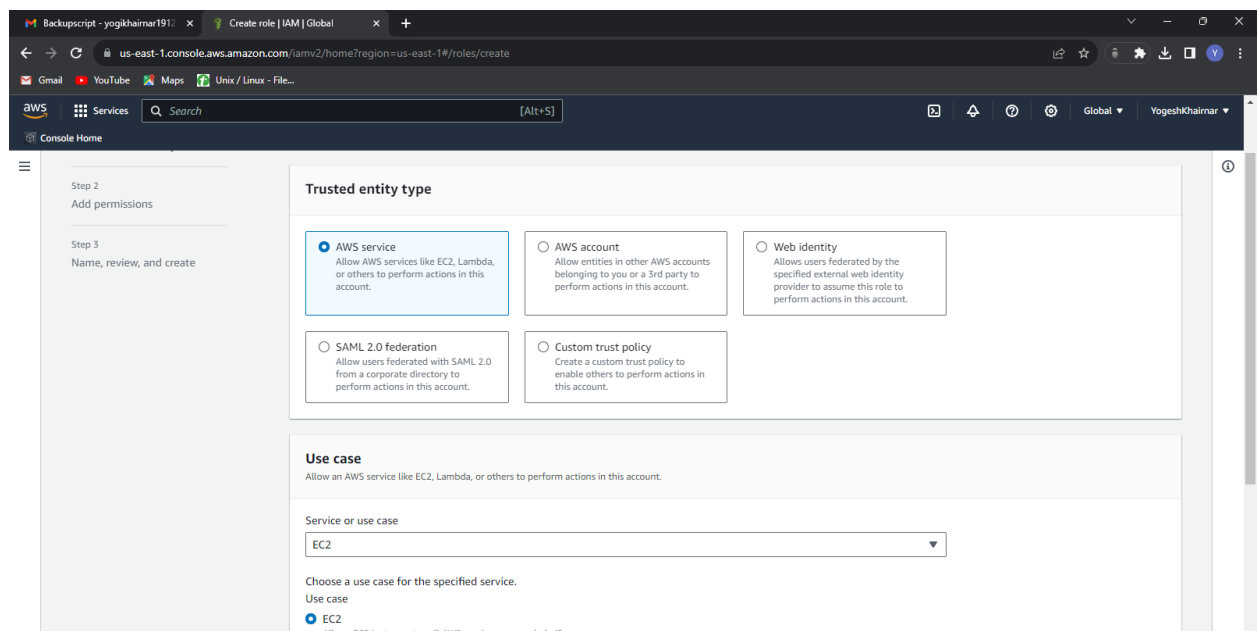


2. IAM(Role):-Create a role.

Search for IAM>Role>Create role.



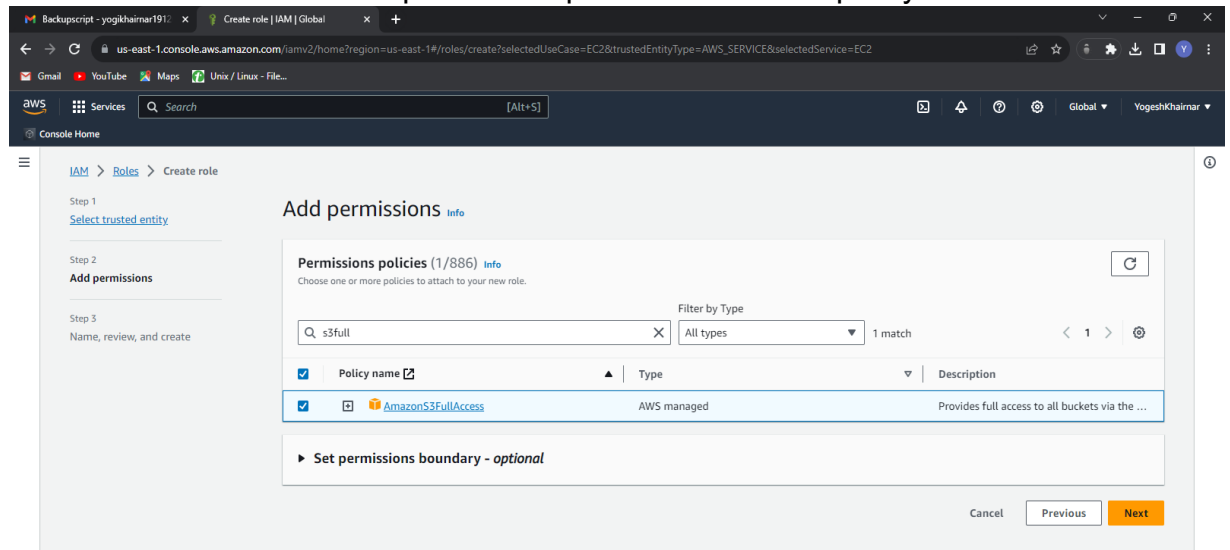
Choose use case as EC2 and keep rest of settings as it is(default)>Next.



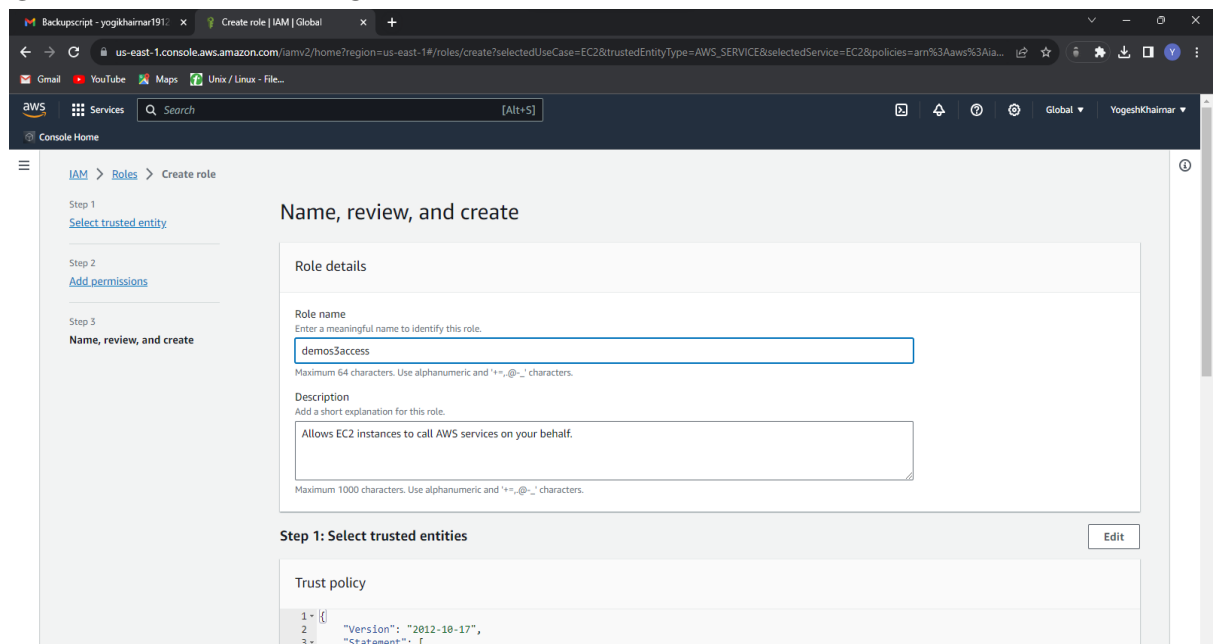
Name :- Yogesh Dilip Khairnar

College:-B.K.Birla College

Search for S3FullAccess in permission policies select that policy>Next.

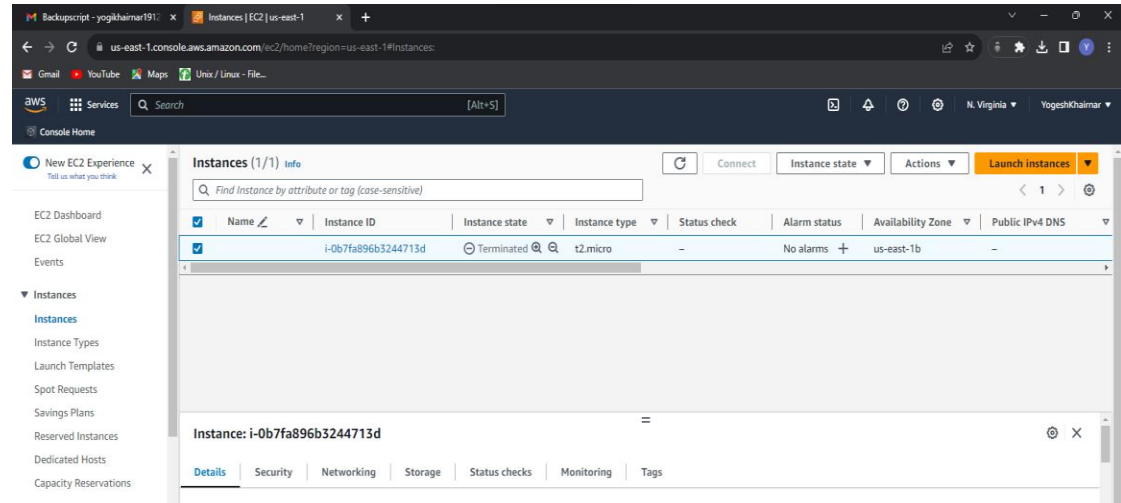


Give the name of role>Create role.

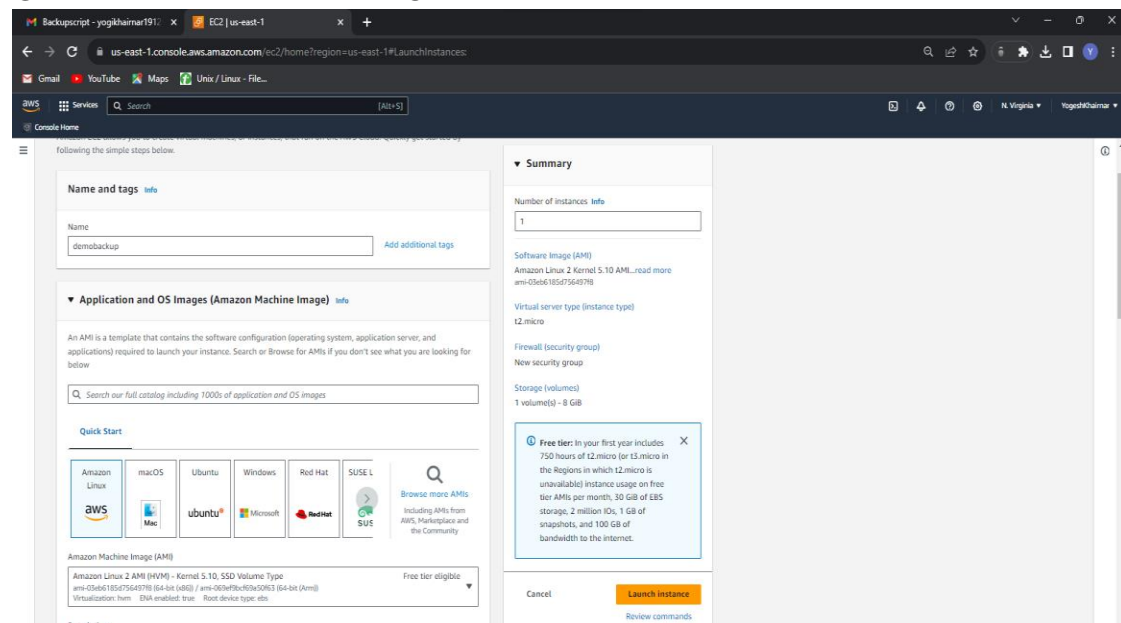


3. EC2:-Launch EC2 Instance.

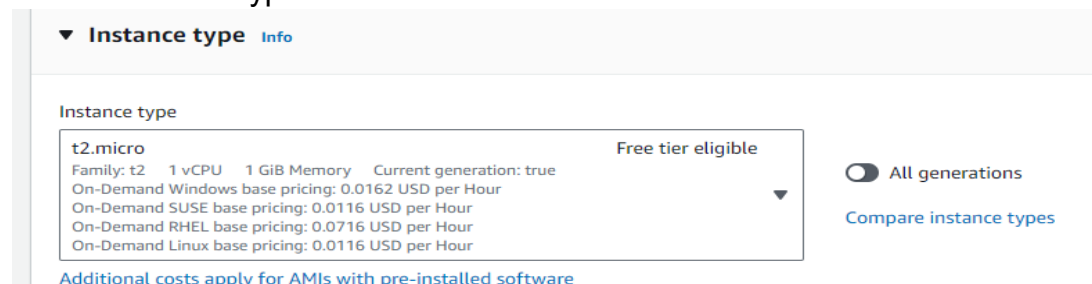
Search for EC2>Instance>Launch Instance.



Give the name of instance>Choose AMI as Amazon Linux2



Select Instance type as t2.micro.



Create new key pair>Give key pair name>select key pair type as.ppk>Create.

▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select

Create new key pair

AWS Services Search [Alt+S] N. Virginia YogeshKhairnar

Console Home

On-demand Linux (Amazon Linux 2) 100% per hour
On-demand Linux (Ubuntu 18.04) 100% per hour
On-demand Linux (CentOS 7) 100% per hour

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select

▼ Network settings Info

Network info
vpc-6387960d4ca0311

Subnet info
No preference (Default subnet in any availability zone)

Auto-assign public IP info
Enable

Create key pair

Key pair name
Key pairs allow you to connect to your instance securely.

demokey

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ RSA
RSA encrypted private and public key pair

☐ ED25519
ED25519 encrypted private and public key pair

Private key file format

☐ .pem
For use with OpenSSH

☒ .ppk
For use with PuTTY

When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#)

Cancel Create key pair

Advanced details>Select the role which you created>Launch Instance.

Backupscript - yogkhairnar191: x EC2 | us-east-1

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances

Amazon Services Search [Alt+S] N. Virginia YogeshKhairnar

Console Home

U x File systems Edit

▼ Advanced details Info

Purchasing option Info
☐ Request Spot Instances

Domain join directory Info
Select

Create new directory

IAM instance profile Info
Select

Q |

Select

demo3access

arn:aws:iam::680619236094:instance-profile/demo3access

Create new IAM profile

▼ Summary

Number of instances Info
1

Firewall (security group)
New security group

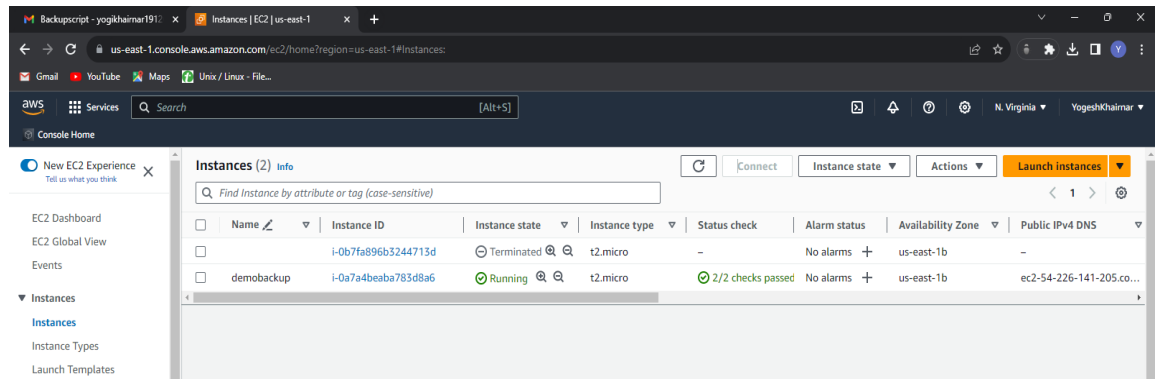
Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOPS, 1 TB of snapshots

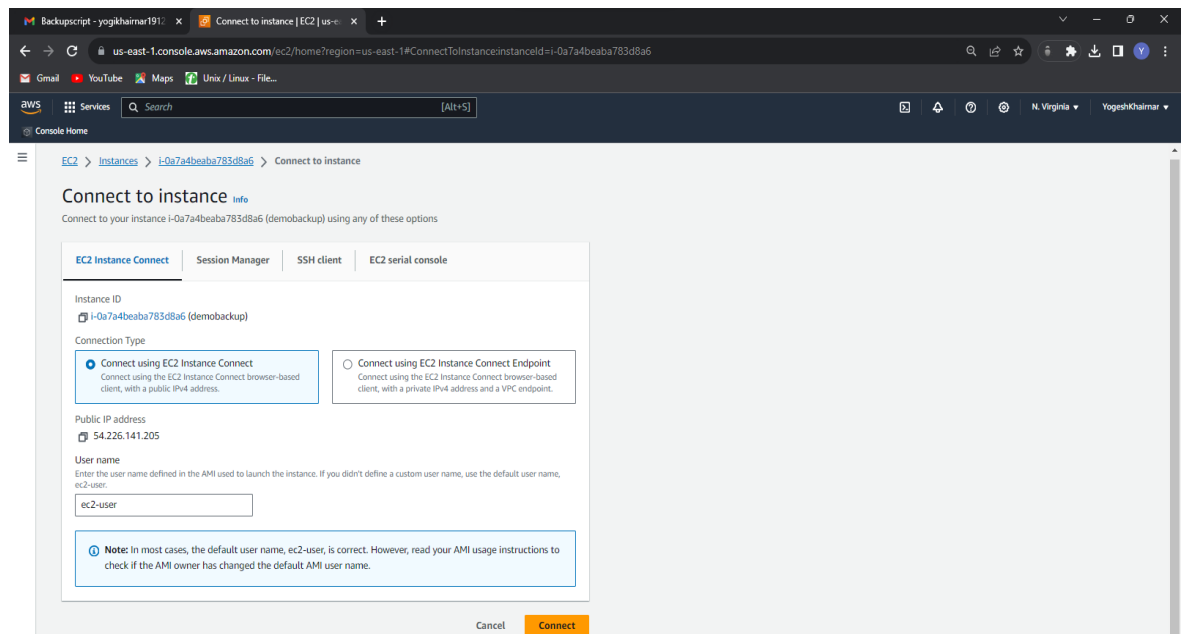
Name :- Yogesh Dilip Khairnar

College:-B.K.Birla College

Wait until the status check pass gets 2/2 once status check gets passed then connect it.



EC2 Instance connect>Connect.

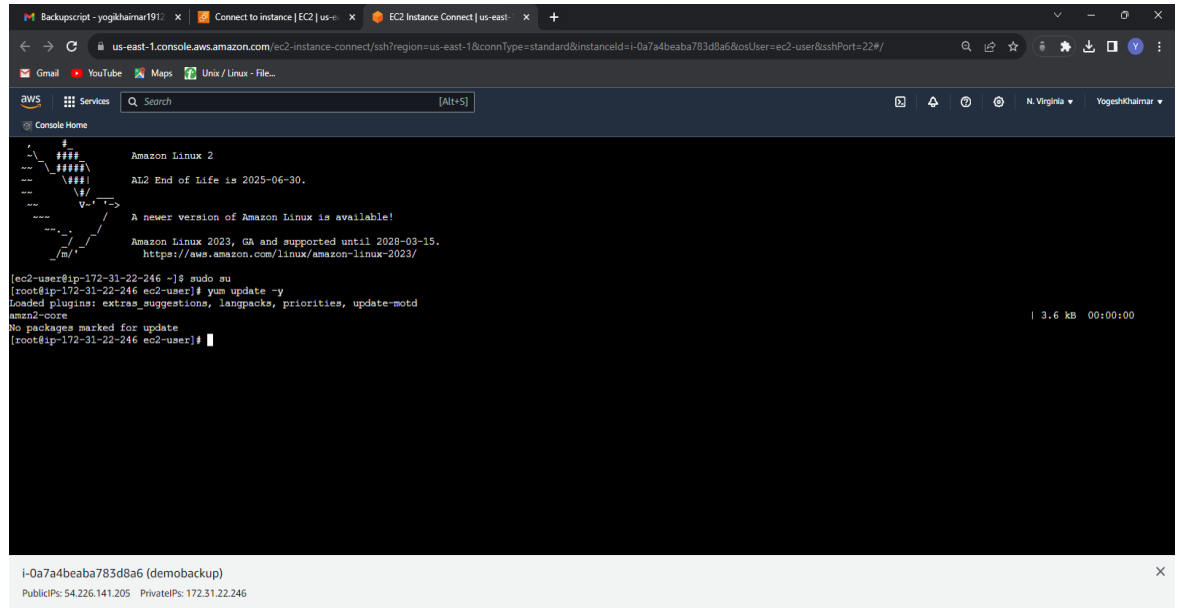


4. Installation of MongoDB and creating a database.

Linux terminal will get open when you connect the instance>write the following commands

`sudo su` (Enter to root user)

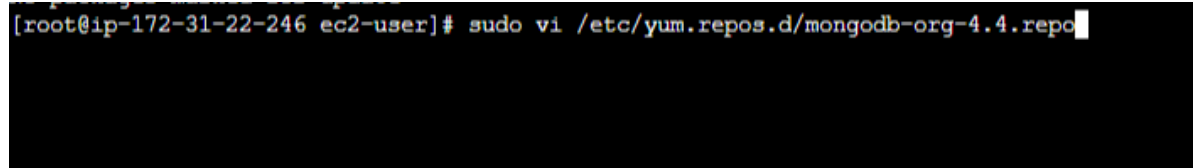
`yum update -y` (Will update the new dependency if available)



```
Amazon Linux 2
AL2 End of Life is 2025-06-30.
A newer version of Amazon Linux is available!
Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-22-246 ~]$ sudo su
[root@ip-172-31-22-246 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amazon2-core
No packages marked for update
[root@ip-172-31-22-246 ec2-user]#
```

`sudo vi /etc/yum.repos.d/mongodb-org-4.4.repo` (Vim editor will get open)



```
[root@ip-172-31-22-246 ec2-user]# sudo vi /etc/yum.repos.d/mongodb-org-4.4.repo
```

In vim editor press I write the following

[mongodb-org-4.4]

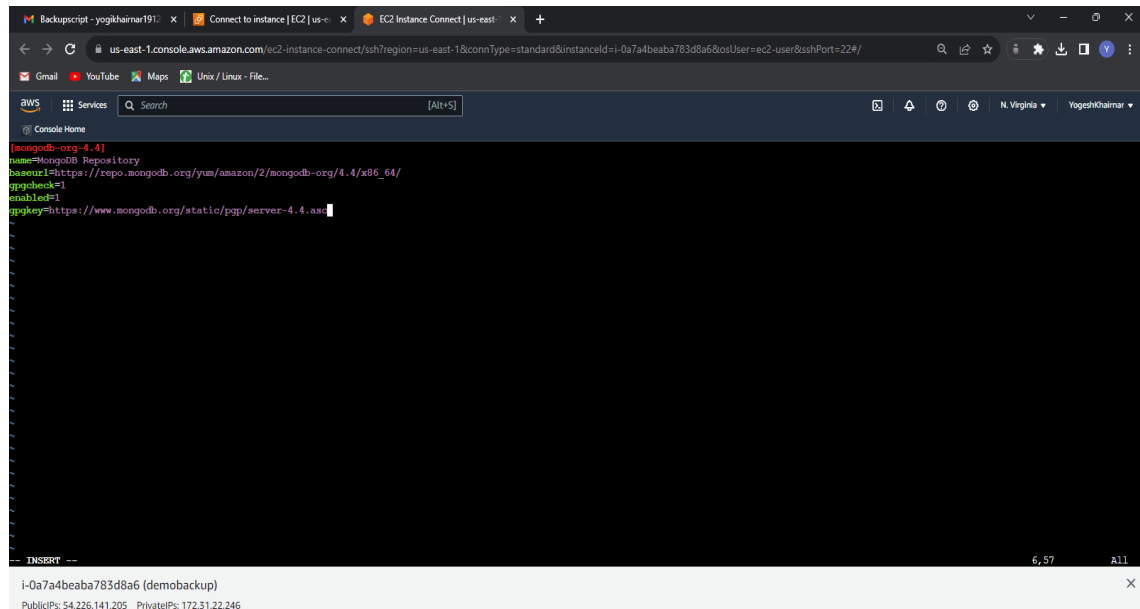
name=MongoDB Repository

baseurl=https://repo.mongodb.org/yum/amazon/2/mongodb-org/4.4/x86_64/

gpgcheck=1

enabled=1

gpgkey=https://www.mongodb.org/static/pgp/server-4.4.asc

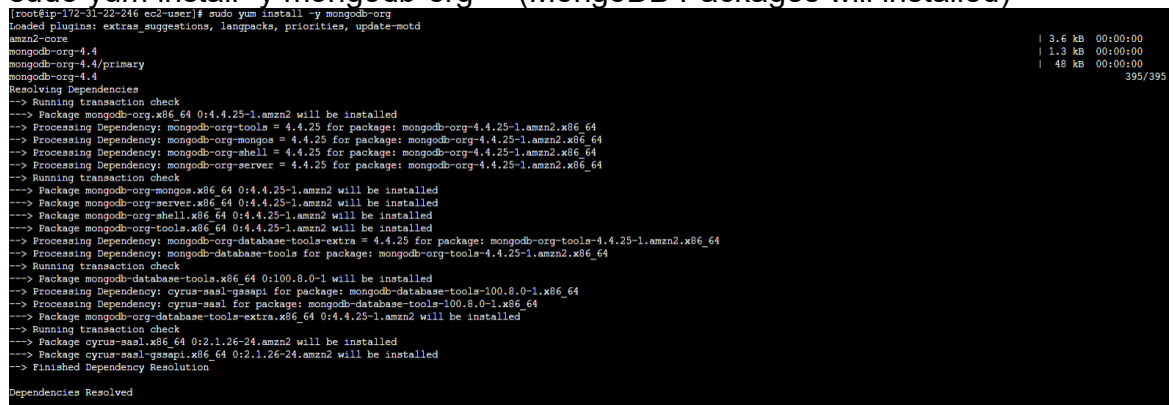


```
[mongodb-org-4.4]
name=MongoDB Repository
baseurl=https://repo.mongodb.org/yum/amazon/2/mongodb-org/4.4/x86_64/
gpgcheck=1
enabled=1
gpgkey=https://www.mongodb.org/static/pgp/server-4.4.asc

```

After writing save it by pressing **Esc- shift + : - wq**

sudo yum install -y mongodb-org (MongoDB Packages will be installed)

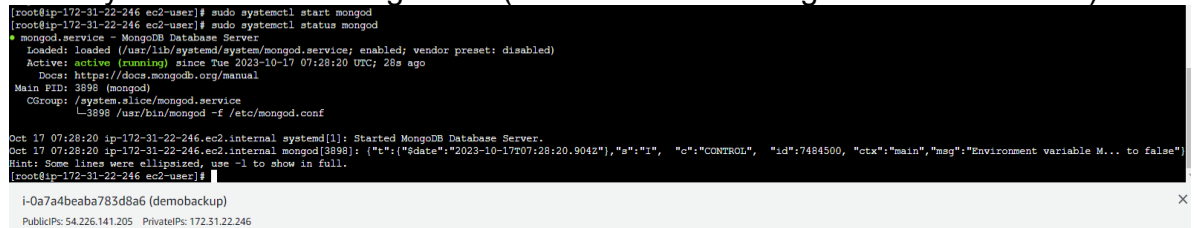


```
[root@ip-172-31-22-246 ec2-user]# sudo yum install -y mongodb-org
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
mongodb-org-4.4                                | 3.6 kB  00:00:00
mongodb-org-4.4/primary                        | 1.3 kB  00:00:00
mongodb-org-4.4                               | 48 kB  00:00:00
395/395
Resolving Dependencies
--> Running transaction check
--> Package mongodb-org.x86_64 0:4.4.25-1.amzn2 will be installed
--> Processing Dependency: mongodb-org-tools = 4.4.25 for package: mongodb-org-4.4.25-1.amzn2.x86_64
--> Processing Dependency: mongodb-org-mongos = 4.4.25 for package: mongodb-org-4.4.25-1.amzn2.x86_64
--> Processing Dependency: mongodb-org-shell = 4.4.25 for package: mongodb-org-4.4.25-1.amzn2.x86_64
--> Processing Dependency: mongodb-org-server = 4.4.25 for package: mongodb-org-4.4.25-1.amzn2.x86_64
--> Running transaction check
--> Package mongodb-org-mongos.x86_64 0:4.4.25-1.amzn2 will be installed
--> Package mongodb-org-server.x86_64 0:4.4.25-1.amzn2 will be installed
--> Package mongodb-org-shell.x86_64 0:4.4.25-1.amzn2 will be installed
--> Package mongodb-org-tools.x86_64 0:4.4.25-1.amzn2 will be installed
--> Processing Dependency: mongodb-org-database-tools-extra = 4.4.25 for package: mongodb-org-tools-4.4.25-1.amzn2.x86_64
--> Processing Dependency: mongodb-org-database-tools for package: mongodb-org-tools-4.4.25-1.amzn2.x86_64
--> Running transaction check
--> Package mongodb-database-tools.x86_64 0:100.8.0-1 will be installed
--> Processing Dependency: cyrus-sasl-gssapi for package: mongodb-database-tools-100.8.0-1.x86_64
--> Processing Dependency: cyrus-sasl for package: mongodb-database-tools-100.8.0-1.x86_64
--> Package mongodb-org-database-tools-extra.x86_64 0:4.4.25-1.amzn2 will be installed
--> Running transaction check
--> Package cyrus-sasl.x86_64 0:2.1.26-24.amzn2 will be installed
--> Package cyrus-sasl-gssapi.x86_64 0:2.1.26-24.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved
```

sudo systemctl start mongod (MongoDB will start)

sudo systemctl status mongod (It will show that mongod is active or not)



```
[root@ip-172-31-22-246 ec2-user]# sudo systemctl start mongod
[root@ip-172-31-22-246 ec2-user]# sudo systemctl status mongod
* mongod.service - MongoDB Database Server
   Loaded: loaded (/usr/lib/systemd/system/mongod.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2023-10-17 07:28:20 UTC; 28s ago
     Docs: https://docs.mongodb.org/manual
   Main PID: 3898 (mongod)
   CGroup: /system.slice/mongod.service
           └─3898 /usr/bin/mongod -f /etc/mongod.conf

Oct 17 07:28:20 ip-172-31-22-246.ec2.internal systemd[1]: Started MongoDB Database Server.
Oct 17 07:28:20 ip-172-31-22-246.ec2.internal mongod[3898]: {"t":{"$date":"2023-10-17T07:28:20.904Z"},"s":"i", "c":"CONTROL", "id":7484500, "ctx":"main","msg":"Environment variable M... to false"}
Hint: Some lines were ellipsized, use -l to show in full.
[root@ip-172-31-22-246 ec2-user]#
```

Now go to Mongo terminal and create a database.

mongo (mongo terminal will open)

```
[root@ip-172-31-22-246 ec2-user]# mongo
MongoDB shell version v4.4.25
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("690231c0-8c31-419c-af41-e90232a36fd7") }
MongoDB server version: 4.4.25
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
  https://docs.mongodb.com/
Questions? Try the MongoDB Developer Community Forums
  https://community.mongodb.com
---
The server generated these startup warnings when booting:
  2023-10-17T07:28:22.028+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
  ---
>
```

show dbs (It will show the database which are present)

use mybackupDB (Database will be created,you can use other name instead of mybackupDB)

db.createCollection("demoCollection") (Collection of database will be created)

exit

```
> show dbs
admin    0.000GB
config  0.000GB
local   0.000GB
> use mybackupDB
switched to db mybackupDB
> db.createCollection("demoCollection")
{ "ok" : 1 }
> exit
```

mkdir DBbackups (Directory will be created,you can replace DBbackups it by other name)

sudo vi backupfile.sh (vim editor will open and .sh file will be created)

```
[root@ip-172-31-22-246 ec2-user]# mkdir DBbackups
[root@ip-172-31-22-246 ec2-user]# sudo vi backupfile.sh
```

Editor will open press I write the following script

#!/bin/bash

Set the backup folder path

folder_name="/home/ec2-user/**your directory name which you created**/"

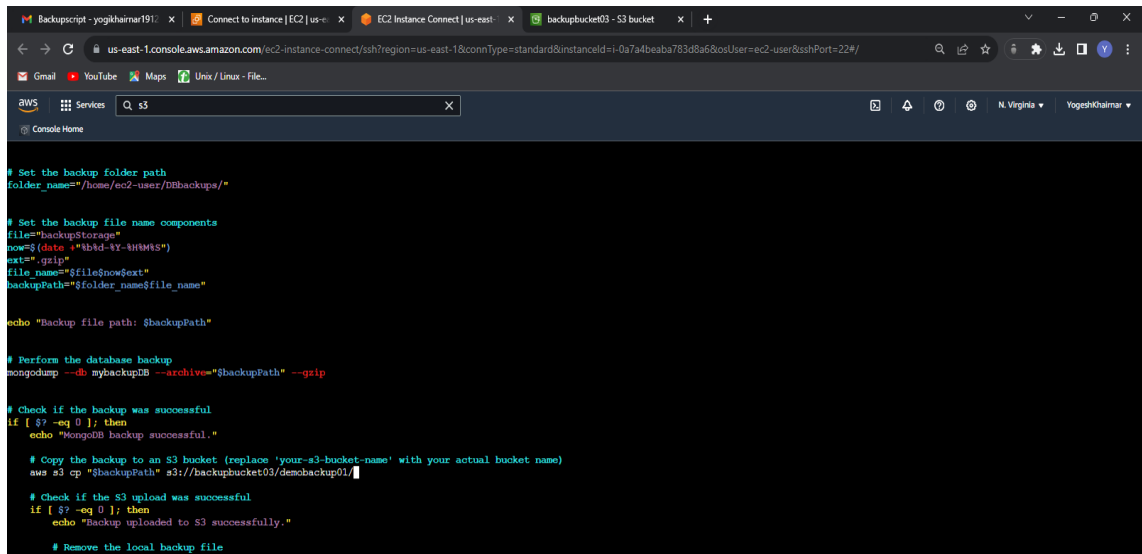
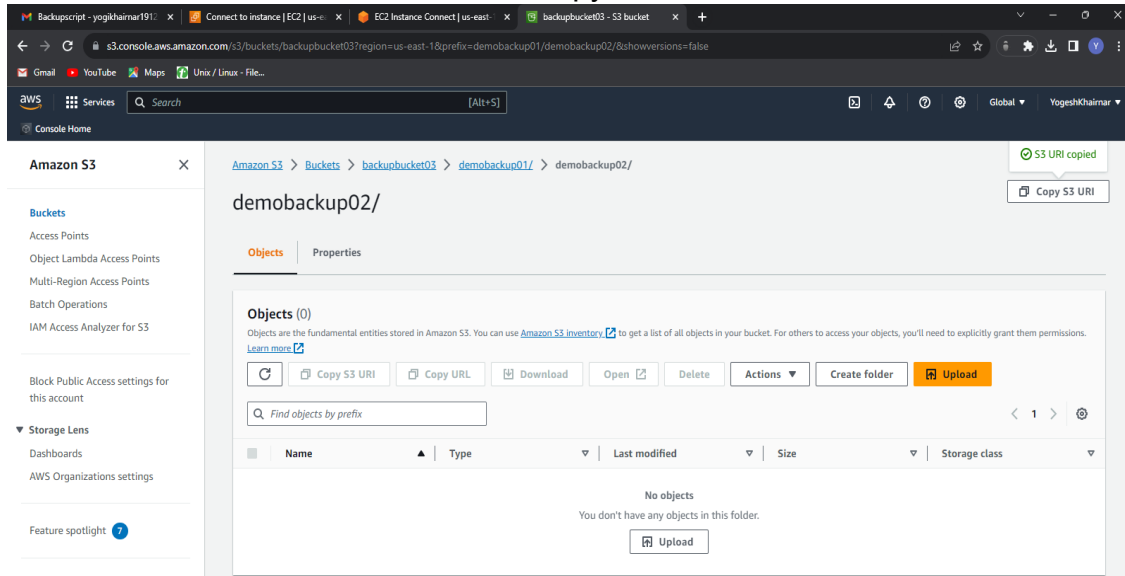
Set the backup file name components

file="backupStorage"

now=\$(date +"%b%d-%Y-%H%M%S")

```
ext=".gzip"
file_name="$file$now$ext"
backupPath="$folder_name$file_name"
echo "Backup file path: $backupPath"
# Perform the database backup
mongodump --db give your database name --archive="$backupPath" --gzip
# Check if the backup was successful
if [ $? -eq 0 ]; then
    echo "MongoDB backup successful."
# Copy the backup to an S3 bucket (replace 'your-s3-bucket-name' with your
actual bucket name)
    aws s3 cp "$backupPath" s3://your-s3-path
# Check if the S3 upload was successful
    if [ $? -eq 0 ]; then
        echo "Backup uploaded to S3 successfully."
# Remove the local backup file
        rm "$backupPath"
        echo "Local backup file removed."
    else
        echo "Error: Failed to upload the backup to S3."
    fi
else
    echo "Error: MongoDB backup failed."
fi
```

for S3 Path-Go to s3>bucket >folder>copy s3url



Save the script by pressing **Esc – shift + : - wq**

chmod 700 backupfile.sh (It will give all permission to file)

./backupfile.sh (It will execute the file)

```
[root@ip-172-31-22-246 ec2-user]# sudo chmod 700 backupfile.sh
[root@ip-172-31-22-246 ec2-user]# sudo ./backupfile.sh
Backup file path: /home/ec2-user/DBbackups/backupStorageOct17-2023-074700.gz
2023-10-17T07:47:00.653+0000   writing mybackupDB.democollection to archive '/home/ec2-user/DBbackups/backupStorageOct17-2023-074700.gz'
2023-10-17T07:47:00.657+0000   done dumping mybackupDB.democollection (0 documents)
MongoDB backup successful.
upload failed: DBbackups/backupStorageOct17-2023-074700.gz to s3://backupbucket03/demobackup01/backupStorageOct17-2023-074700.gz Unable to locate credentials
Error: Failed to upload the backup to S3.
[root@ip-172-31-22-246 ec2-user]#
```

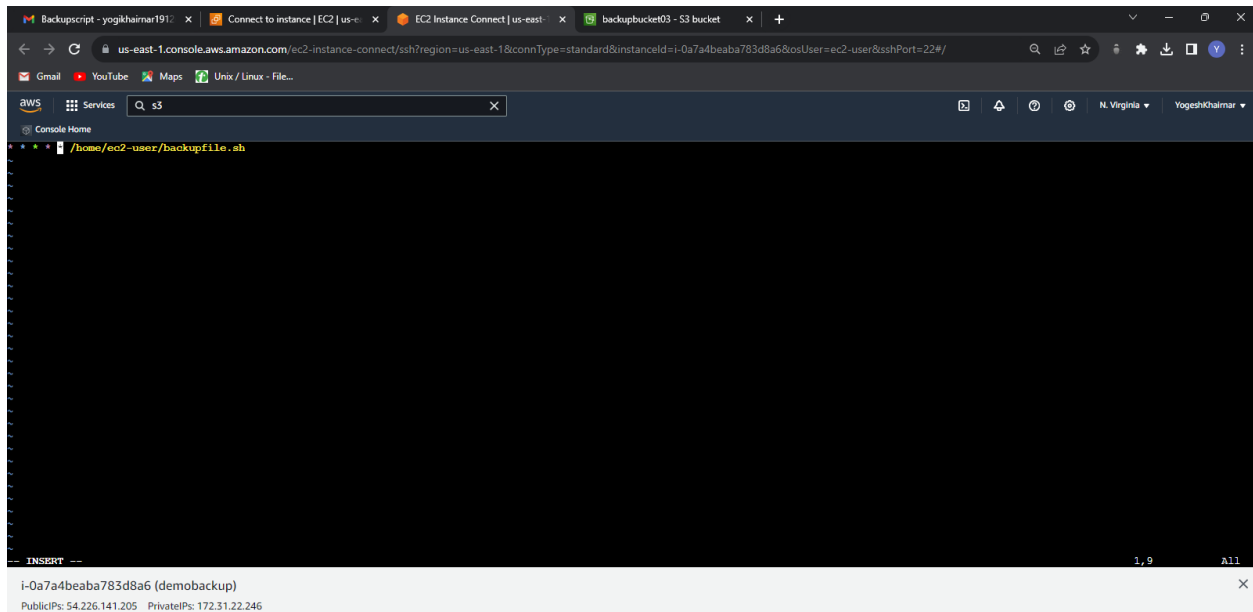
Now we have to schedule the file so that the backup file will be automatically backed up in every one minute

Write the following command

crontab -e

```
[root@ip-172-31-22-246 ec2-user]# crontab -e  
i-0a7a4beaba783d8a6 (demobackup)
```

* * * * * /home/ec2-user/give your backupfile name (i.e .sh file)

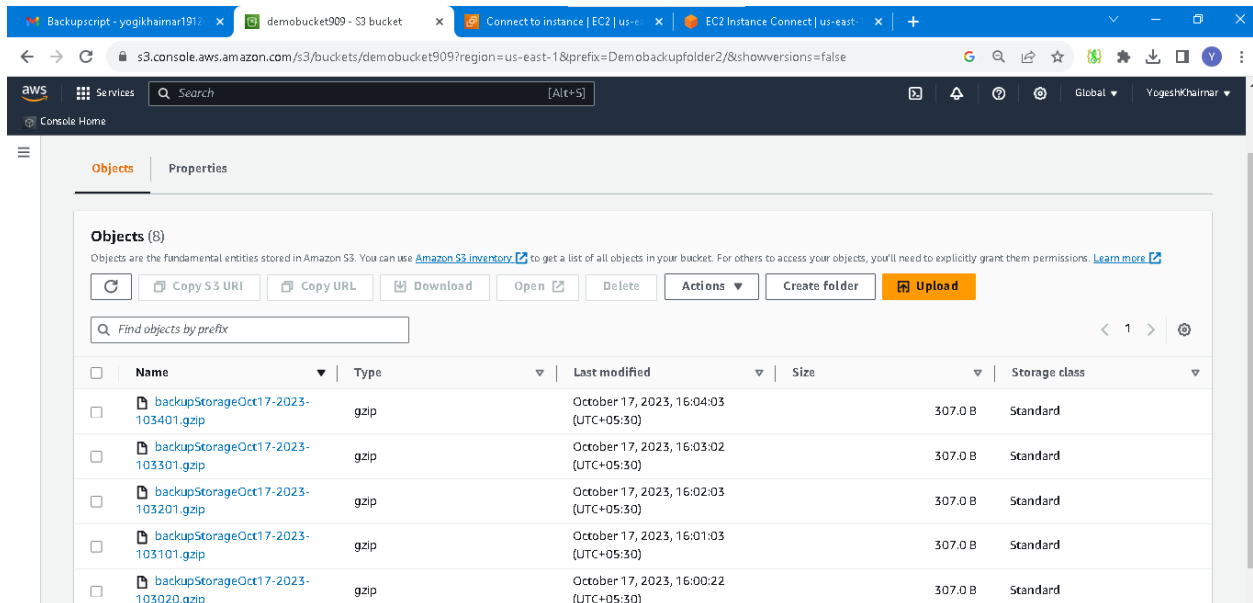


Save it by pressing **Esc – shift + : - wq**

Again execute the file

./backupfile.sh

Go to s3>bucket>folder you will see that the files have been backedup in every 1 min



Hence this can be one of the solution to automate backup of MongoDB database.