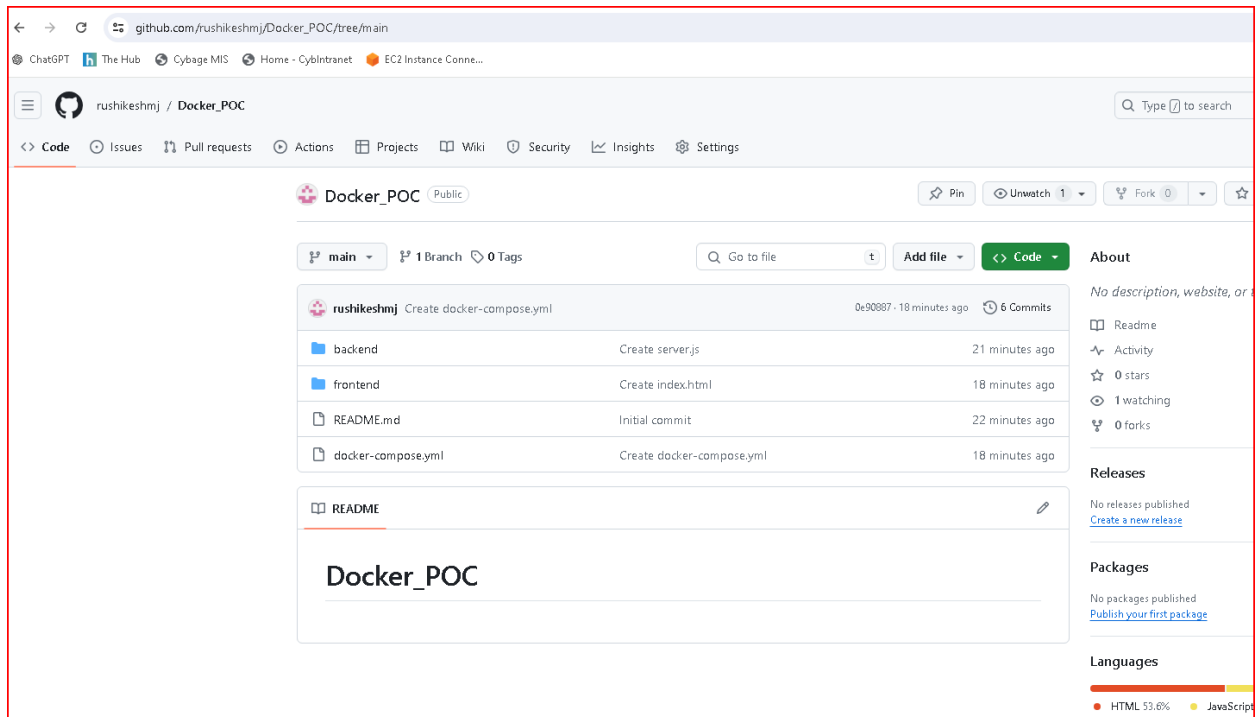


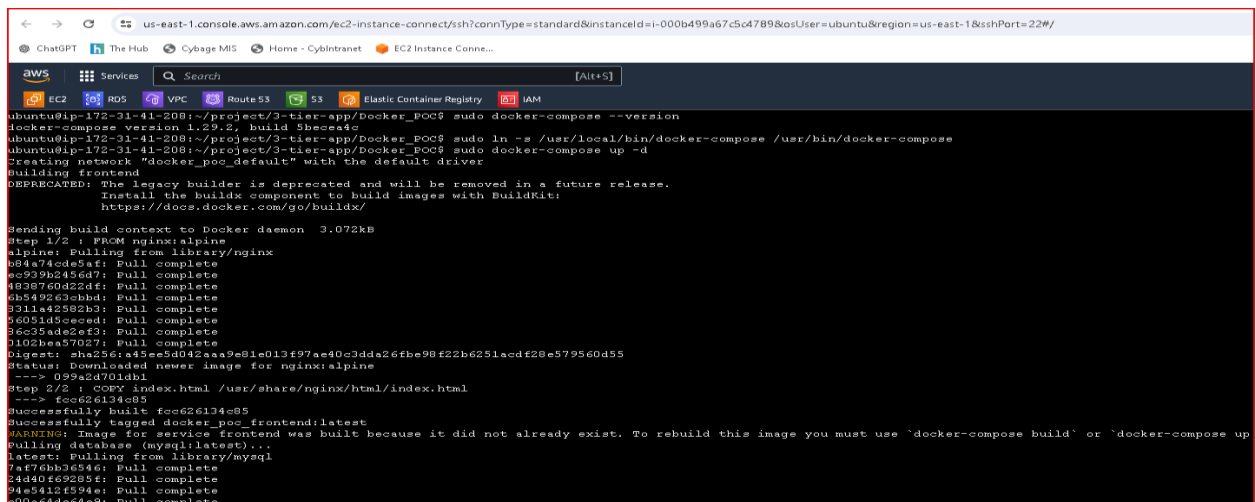
RDS-POC

For a 3-tier application (typically frontend, backend, and database), we can use Docker Compose to orchestrate the deployment

Install docker-compose



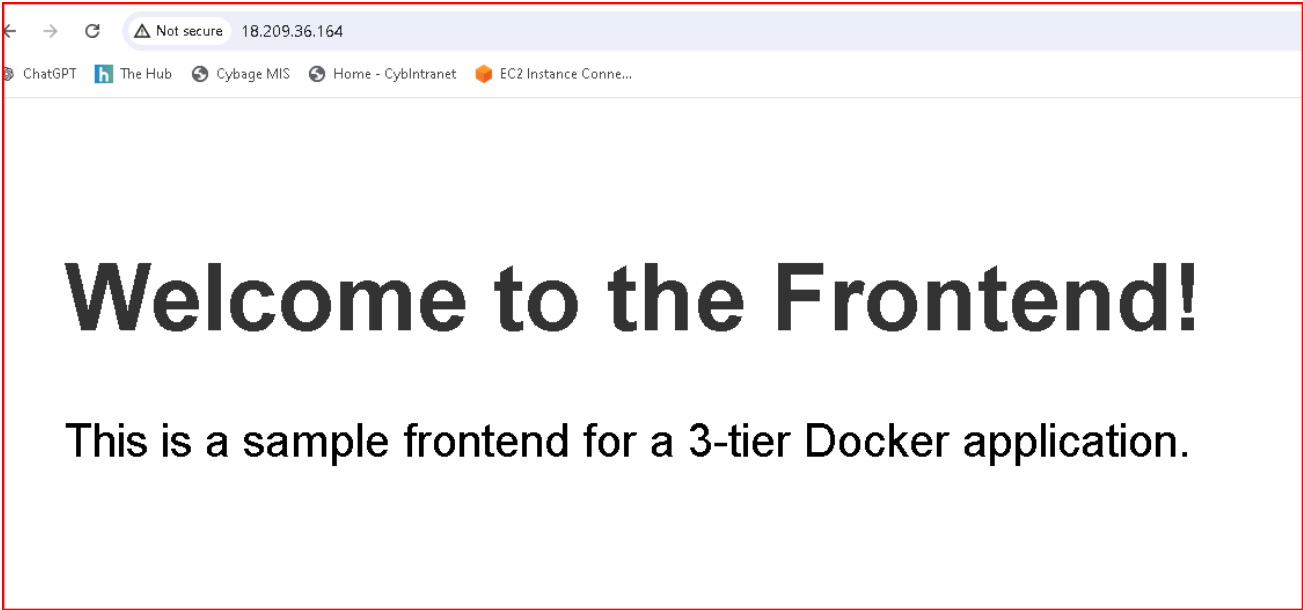
Clone the project in new directory



sudo docker-compose up -d

```
Successfully built 89d676276f01
Successfully tagged docker_poc_backend:latest
WARNING: Image for service backend was built because it did not already exist. To rebuild this image you must use `docker-compose build` or `docker-compose up --build`.
Creating docker_poc_frontend_1 ... done
Creating docker_poc_database_1 ... done
Creating docker_poc_backend_1 ... done
ubuntu@ip-172-31-41-208:~/project/3-tier-app/Docker_POC$ sudo docker ps
Command 'sud' not found, did you mean:
  command 'std' from snap std (1.0.1)
  command 'sup' from deb sup (20100519-3)
  command 'sudo' from deb sudo (1.9.9-1ubuntu2.4)
  command 'sudo' from deb sudo-ldap (1.9.9-1ubuntu2.4)
  command 'sed' from deb sed (4.8-1ubuntu2)
  command 'sbd' from deb sbd (1.5.1-1ubuntu2)
  command 's3d' from deb s3d (0.2.2.1-3build1)
  command 'snd' from deb snd-gui-jack (22.1-1)
  command 'snd' from deb snd-gui-pulse (22.1-1)
  command 'snd' from deb snd-nox (22.1-1)
  command 'sur' from deb subtle (0.11.3224-xi-2.2build4)
  command 'sul' from deb hxttools (20211204-1)
  command 'spd' from deb spd (1.3.0-1ubuntu3)
  command 'su' from deb util-linux (2.37.2-4ubuntu3)
  command 'sum' from deb coreutils (8.32-4.1ubuntu1)
See 'snap info <snapname>' for additional versions.
ubuntu@ip-172-31-41-208:~/project/3-tier-app/Docker_POC$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                                     NAMES
5be9fd076c0a   docker_poc_backend   "docker-entrypoint.s..." 36 seconds ago Up 35 seconds 0.0.0.0:3000->3000/tcp, :::3000->3000/tcp  docker_poc_backend_1
b6a9923c74f1   docker_poc_frontend   "/docker-entrypoint..." 37 seconds ago Up 35 seconds 0.0.0.0:80->80/tcp, :::80->80/tcp        docker_poc_frontend_1
ubuntu@ip-172-31-41-208:~/project/3-tier-app/Docker_POC$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                                     NAMES
```

Frontend



create a bastion host in public subnet in same vpc and accessing rds instance from it

Instances (1) Info									
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>				Running		< 1 > ⚙			
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
<input type="checkbox"/>	Bastion host	i-019e41145a9466052	Running	t2.medium	2/2 checks passed	View alarms	us-east-1a	ec2-3-208-243-50.comp...	3.208.243.50

Create a iam user and provide necessary permissions by attaching policy for taking backup and retention

Create snapshot from aws cli

```
root@ip-172-31-45-241:/home/ubuntu# aws configure set region ap-south-1
root@ip-172-31-45-241:/home/ubuntu# aws rds create-db-snapshot --db-instance-identifier database1 --db-snapshot-identifier rdsbackup
```

Succesfully created Snapshot

```
"DBSnapshot": {
  "DBSnapshotIdentifier": "rdsbackup",
  "DBInstanceIdentifier": "database1",
  "Engine": "mysql",
  "AllocatedStorage": 20,
  "Status": "creating",
  "Port": 3306,
  "AvailabilityZone": "ap-south-1a",
  "VpcId": "vpc-048703349f46a4b96",
  "InstanceCreateTime": "2024-07-17T08:00:06.844000+00:00",
  "MasterUsername": "karan",
  "EngineVersion": "8.0.35",
  "LicenseModel": "general-public-license",
  "SnapshotType": "manual",
  "OptionGroupName": "default:mysql-8-0",
  "PercentProgress": 0,
  "StorageType": "gp2",
  "Encrypted": true,
  "KmsKeyId": "arn:aws:kms:ap-south-1:397995044220:key/82c4871c-6e4f-4b06-bb24-745a1807a418",
```

Snapshots

Manual

System

Shared with me

Public

Backup service

Exports in Amazon S3

Manual snapshots (2)

Actions

Take snapshot

Filter by manual snapshots

<

1

>

<div></div>	Snapshot name	DB instance or cluster	Snapshot creation time
<div></div>	rdsbackup-s3	database1	July 17, 2024, 20:50 (UTC+05:30)
<div></div>	rdsbackup	database1	July 17, 2024, 20:29 (UTC+05:30)

Export the snapshot to my S3 bucket

RDS > Snapshots > rdsbackup > myrdssnap

myrdssnap

Cancel

Export in Amazon S3

Export identifier
myrdssnap

Source
rdsbackup

ARN
arn:aws:rds:ap-south-1:397995044220:snapshot:rdsbackup

Status
Starting

Exported data
All data in the database will be exported (size up to 20 GB).

S3 bucket
s3://karambucket1/RDSSnap/myrdssnap

IAM role
arn:aws:iam::397995044220:role/service-role/s3rds

AWS KMS key
arn:aws:kms:ap-south-1:397995044220:key/729f0e5c-07d7-4852-8ebd-e28f63f46aac

myrdssnap/

Copy S3 URI

Objects

Properties

Objects (3) Info

Copy S3 URI Copy URL Download Open Delete Actions Create folder Upload

Find objects by prefix

	Name	Type	Last modified	Size	Storage class
	export_info_myrdssnap.json	json	July 18, 2024, 11:09:57 (UTC+05:30)	573.0 B	Standard
	export_tables_info_myrdssnap_from_1_to_1.json	json	July 18, 2024, 11:09:57 (UTC+05:30)	953.0 B	Standard

Create database by restoring from S3

S3 source

S3 bucket

Choose the Amazon S3 bucket that contains your database backup files.

rushikesh174563

S3 prefix - optional Info

Enter the file path prefix for the files stored in your Amazon S3 bucket.

Engine options

Engine type Info

Aurora (MySQL Compatible)

MySQL

Engine Version

Aurora MySQL 3.05.2 (compatible with MySQL 8.0.32)

Create the database by restoring the data from s3

Databases (2)

Group resources

Modify

Actions

Restore from S3

Create database

Filter by databases

< 1 >

	DB identifier	Status	Role	Engine	Region & AZ	Size	Recommendations
	database1	Available	Instance	MySQL Community	ap-south-1a	db.t3.micro	1 Informational
	restoredfroms3	Deleting	Instance	MySQL Community	ap-south-1a	db.r6g.large	