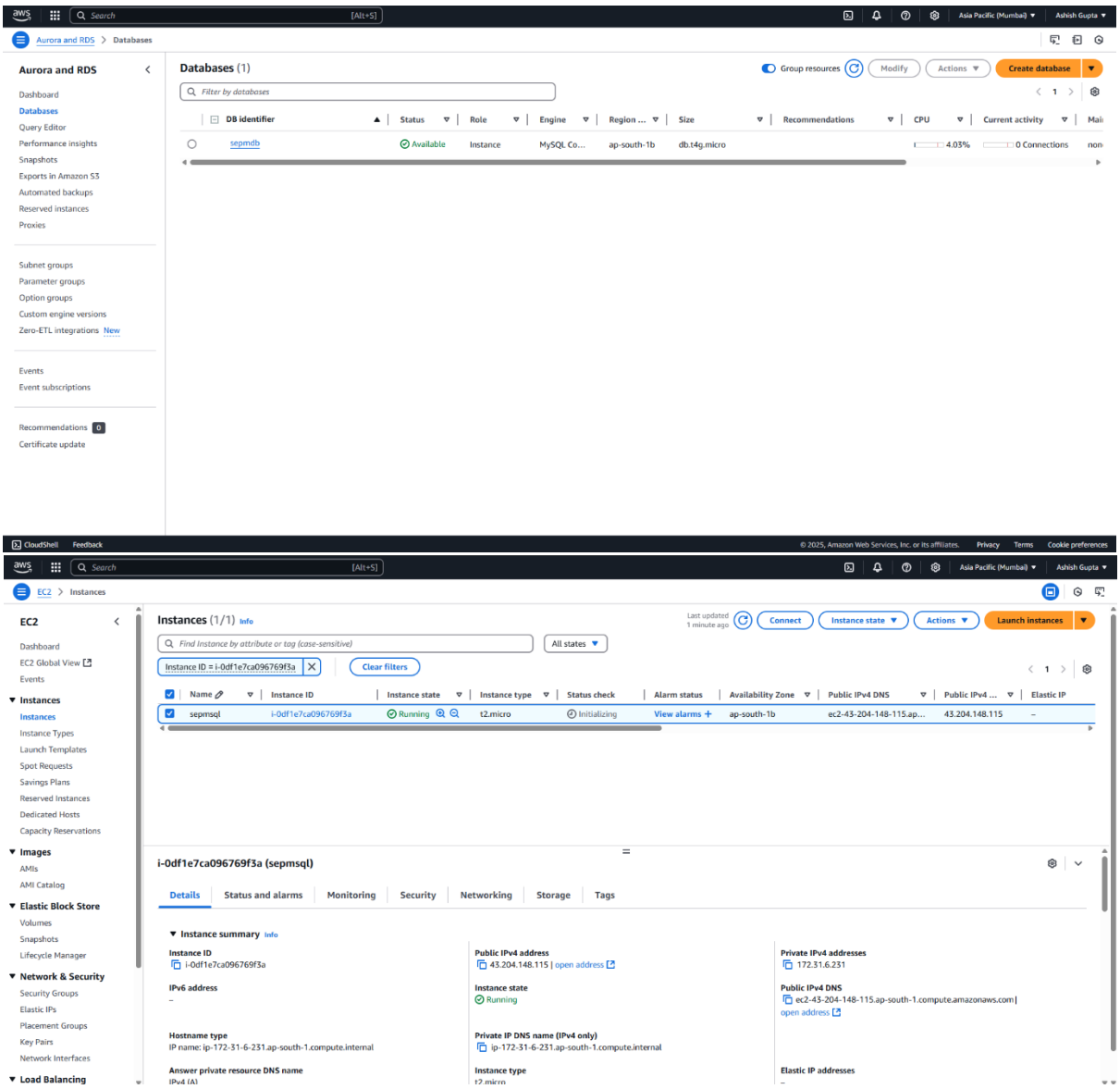
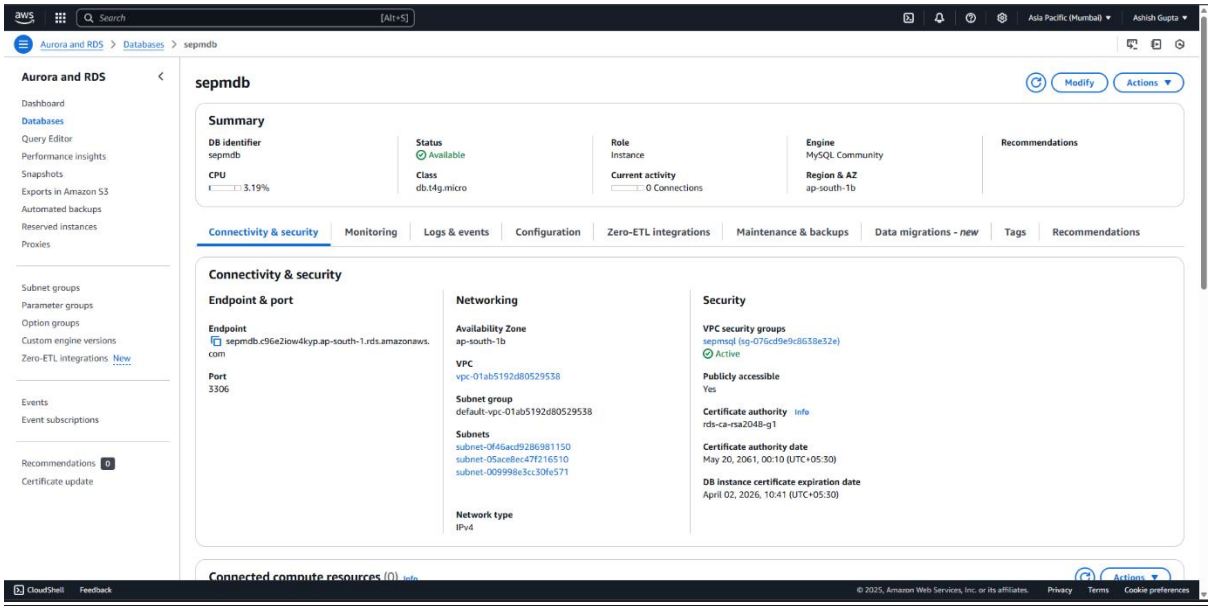
EXPERIMENT 9

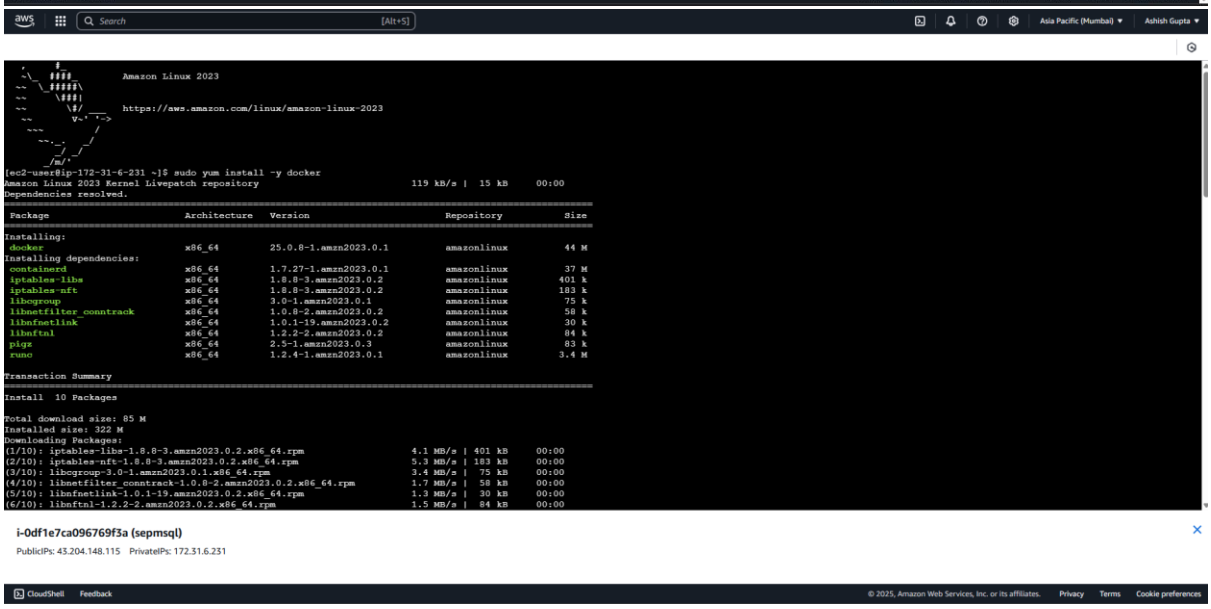
Aim: To understand Docker Architecture and Container Life Cycle, install Docker and execute docker commands to manage images and interact with containers.

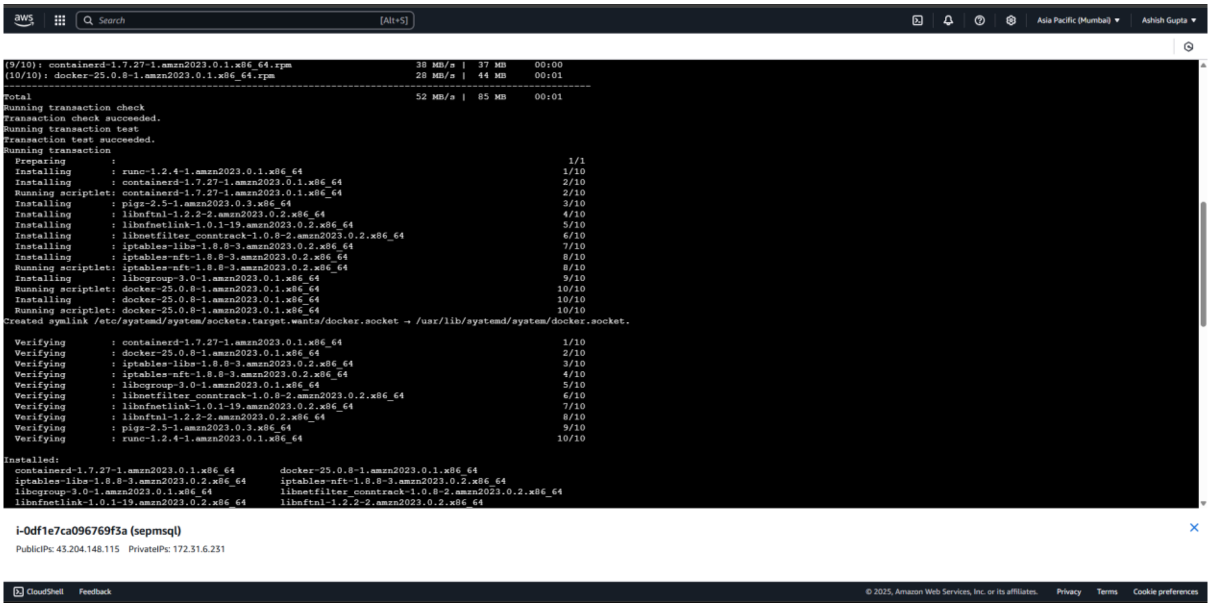
Theory:
Docker is a popular platform that enables developers to build, package, and deploy applications as lightweight, portable, and self-sufficient containers. These containers encapsulate all the necessary dependencies and libraries required for an application to run, ensuring consistency across different environments. Here is a theoretical overview of Docker:
Containerization:
Docker utilizes containerization technology to create isolated environments for applications. Containers are lightweight, standalone, and executable packages that include everything needed to run an application, such as code, runtime, system tools, libraries, and settings. This isolation ensures that applications run consistently across different environments, from development to production.
Docker Engine:
At the core of Docker is the Docker Engine, which is responsible for building, running, and managing containers. It consists of the Docker daemon, which manages containers, images, networks, and volumes, and the Docker client, which allows users to interact with the daemon through the Docker API.
Docker Images:
Docker images are read-only templates used to create containers. They contain the application code, runtime, libraries, dependencies, and other files needed to run the application. Images are built using Dockerfiles, which are text files that define the steps needed to create the image.
Docker Containers:
Containers are instances of Docker images that are running as isolated processes on a host machine. They are lightweight, portable, and can be easily started, stopped, moved, and deleted. Containers provide a consistent environment for applications to run, regardless of the underlying infrastructure.
Benefits of Docker:
Portability: Docker containers can run on any platform that supports Docker, making it easy to deploy applications across different environments.
Efficiency: Containers share the host OS kernel, reducing overhead and improving resource utilization.
Isolation: Containers provide a level of isolation that helps prevent conflicts between applications and dependencies.
Scalability: Docker enables easy scaling of applications by quickly spinning up additional containers.
Consistency: Docker ensures that applications run the same way in development, testing, and production environments.

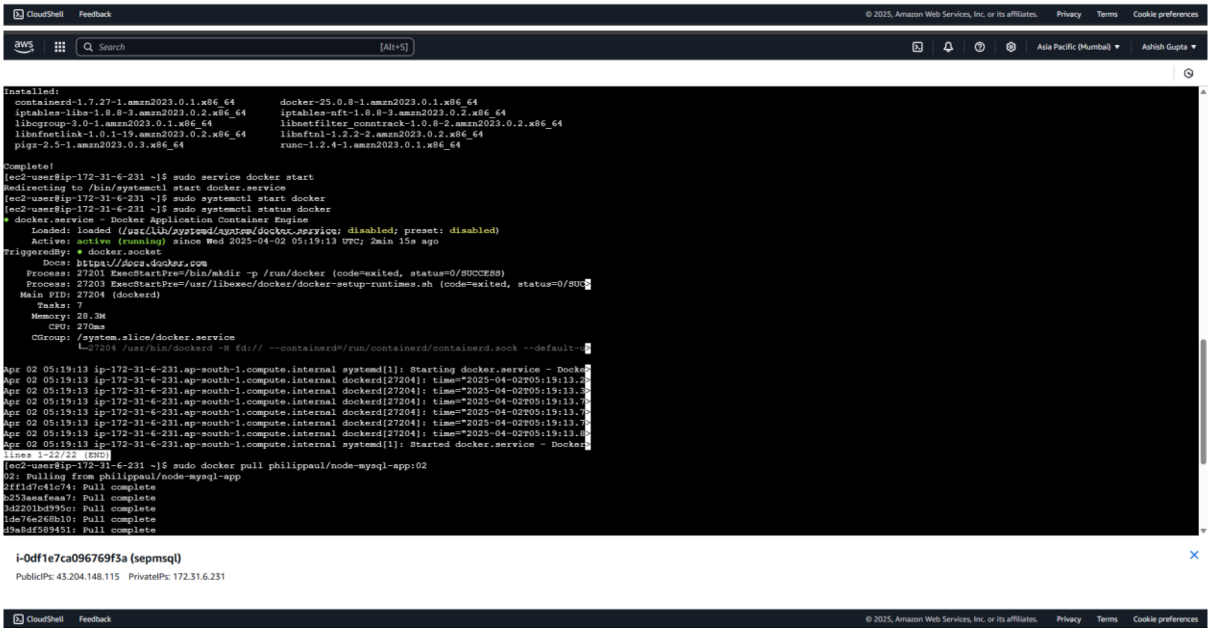
Output:

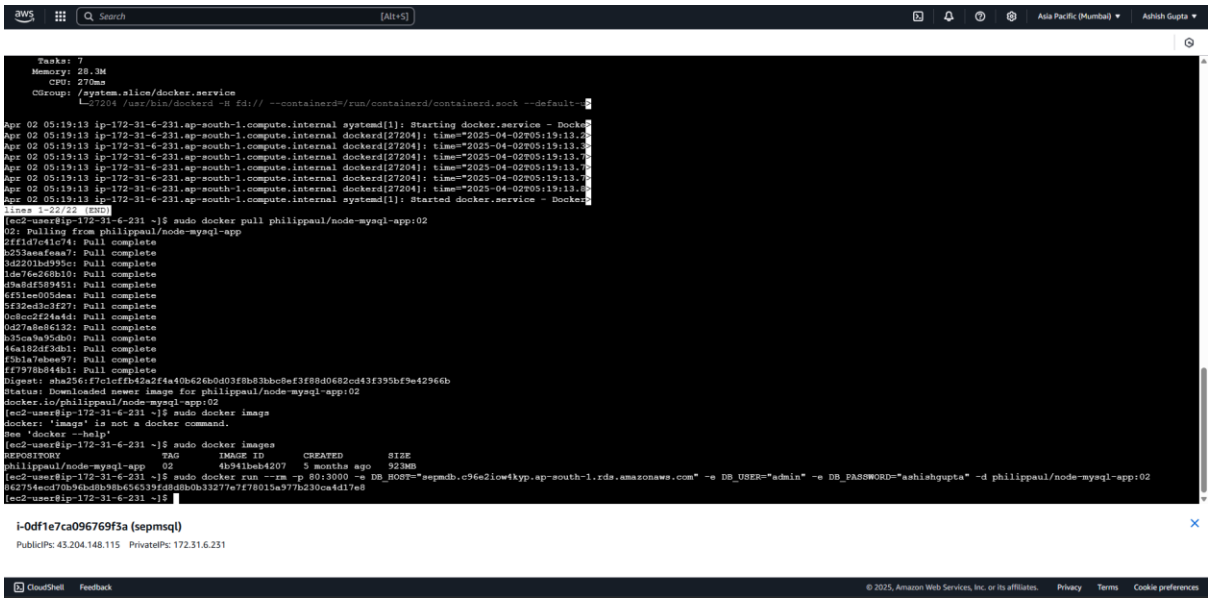

Aurora and RDS
Databases (1)
Create database
sepmdb
Available
db.t4g.micro
EC2
Instances (1/1)
Launch instances
sepmsql
i-0df1e7ca096769f3a
Running
t2.micro
i-0df1e7ca096769f3a (sepmsql)
Instance summary
43.204.148.115
172.31.6.231
ec2-43-204-148-115.ap-south-1.compute.amazonaws.com

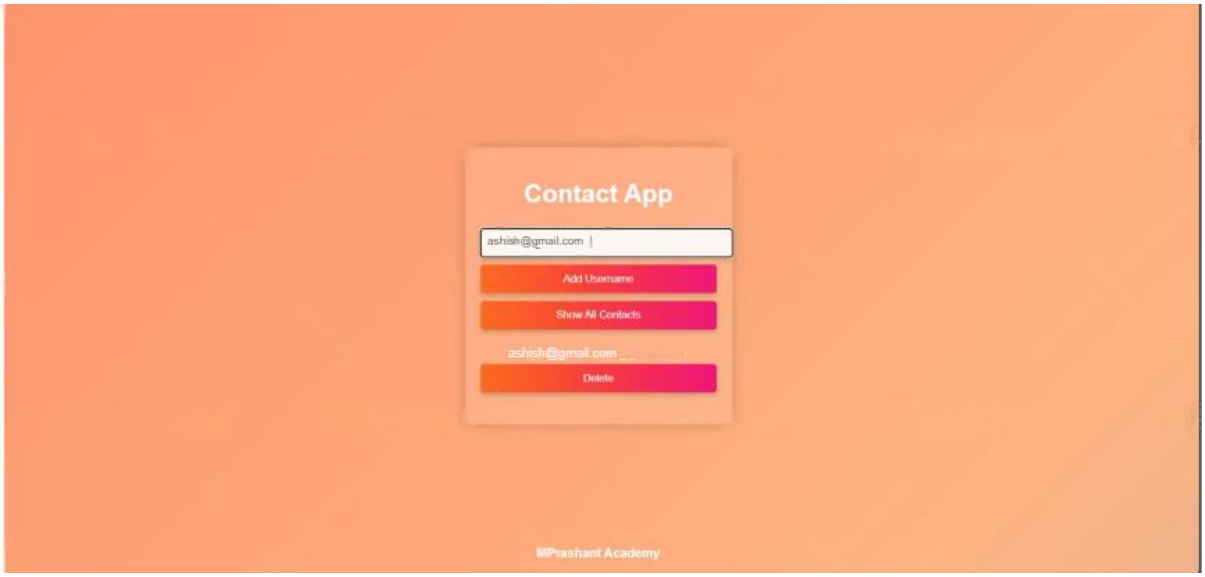

Aurora and RDS > Databases > sepmdb
Aurora and RDS
Dashboard
Databases
Query Editor
Performance insights
Snapshots
Exports in Amazon S3
Automated backups
Reserved instances
Proxies
Subnet groups
Parameter groups
Option groups
Custom engine versions
Zero-ETL integrations New
Events
Event subscriptions
Recommendations 0
Certificate update
sepmdb
Modify
Actions
Summary
DB identifier
sepmdb
Status
Available
Role
Instance
Engine
MySQL Community
Recommendations
CPU
3.19%
Class
db.t4g.micro
Current activity
0 Connections
Region & AZ
ap-south-1b
Connectivity & security
Monitoring
Logs & events
Configuration
Zero-ETL integrations
Maintenance & backups
Data migrations - new
Tags
Recommendations
Connectivity & security
Endpoint & port
Endpoint
sepmdb.c96e2iow4kyp.ap-south-1.rds.amazonaws.com
Port
3306
Networking
Availability Zone
ap-south-1b
VPC
vpc-01ab5192d80529538
Subnet group
default-vpc-01ab5192d80529538
Subnets
subnet-0f46acd9286981150
subnet-05ace8ec47f216510
subnet-009998e3cc30fe571
Network type
IPv4
Security
VPC security groups
sepmsql (sg-076cd9e9c8638e32e)
Active
Publicly accessible
Yes
Certificate authority Info
rds-ca-rsa2048-g1
Certificate authority date
May 20, 2061, 00:10 (UTC+05:30)
DB instance certificate expiration date
April 02, 2026, 10:41 (UTC+05:30)
Connected compute resources (0) Info
Actions
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

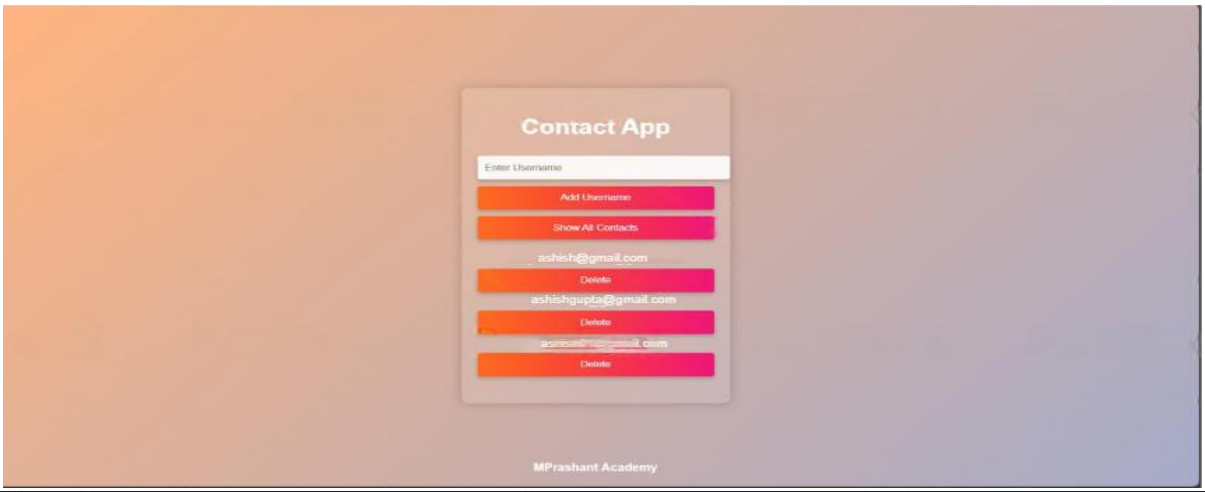

Search
[Alt+S]
Asia Pacific (Mumbai)
Ashish Gupta
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
[ec2-user@ip-172-31-6-231 ~]$ sudo yum install -y docker
Amazon Linux 2023 Kernel Livepatch repository 119 kB/s | 15 kB 00:00
Dependencies resolved.
Package Architecture Version Repository Size
Installing:
docker x86_64 25.0.8-1.amzn2023.0.1 amazonlinux 44 M
Installing dependencies:
containerd x86_64 1.7.27-1.amzn2023.0.1 amazonlinux 37 M
iptables-libs x86_64 1.8.8-3.amzn2023.0.2 amazonlinux 401 k
iptables-nft x86_64 1.8.8-3.amzn2023.0.2 amazonlinux 183 k
libcgroup x86_64 3.0-1.amzn2023.0.1 amazonlinux 75 k
libnetfilter_conntrack x86_64 1.0.8-2.amzn2023.0.2 amazonlinux 58 k
libnfnetlink x86_64 1.0.1-19.amzn2023.0.2 amazonlinux 30 k
libnftnl x86_64 1.2.2-2.amzn2023.0.2 amazonlinux 84 k
pigz x86_64 2.5-1.amzn2023.0.3 amazonlinux 83 k
runc x86_64 1.2.4-1.amzn2023.0.1 amazonlinux 3.4 M
Transaction Summary
Install 10 Packages
Total download size: 85 M
Installed size: 322 M
Downloading Packages:
(1/10): iptables-libs-1.8.8-3.amzn2023.0.2.x86_64.rpm 4.1 MB/s | 401 kB 00:00
(2/10): iptables-nft-1.8.8-3.amzn2023.0.2.x86_64.rpm 5.3 MB/s | 183 kB 00:00
(3/10): libcgroup-3.0-1.amzn2023.0.1.x86_64.rpm 3.4 MB/s | 75 kB 00:00
(4/10): libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64.rpm 1.7 MB/s | 58 kB 00:00
(5/10): libnfnetlink-1.0.1-19.amzn2023.0.2.x86_64.rpm 1.3 MB/s | 30 kB 00:00
(6/10): libnftnl-1.2.2-2.amzn2023.0.2.x86_64.rpm 1.5 MB/s | 84 kB 00:00
i-0df1e7ca096769f3a (sepmsql)
PublicIPs: 43.204.148.115 PrivateIPs: 172.31.6.231
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

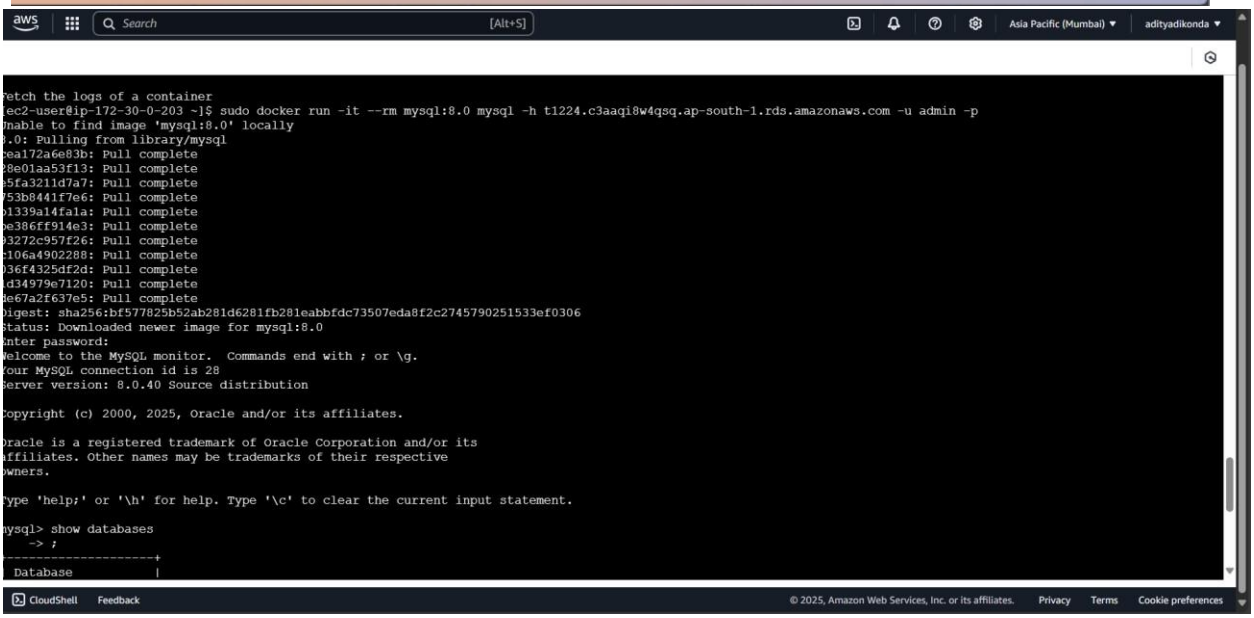

Search
[Alt+S]
Asia Pacific (Mumbai)
Ashish Gupta
(9/10): containerd-1.7.27-1.amzn2023.0.1.x86_64.rpm 38 MB/s | 37 MB 00:00
(10/10): docker-25.0.8-1.amzn2023.0.1.x86_64.rpm 28 MB/s | 44 MB 00:01
Total 52 MB/s | 85 MB 00:01
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing : 1/1
Installing : runc-1.2.4-1.amzn2023.0.1.x86_64 1/10
Installing : containerd-1.7.27-1.amzn2023.0.1.x86_64 2/10
Running scriptlet: containerd-1.7.27-1.amzn2023.0.1.x86_64 2/10
Installing : pigz-2.5-1.amzn2023.0.3.x86_64 3/10
Installing : libnftnl-1.2.2-2.amzn2023.0.2.x86_64 4/10
Installing : libnfnetlink-1.0.1-19.amzn2023.0.2.x86_64 5/10
Installing : libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64 6/10
Installing : iptables-libs-1.8.8-3.amzn2023.0.2.x86_64 7/10
Installing : iptables-nft-1.8.8-3.amzn2023.0.2.x86_64 8/10
Running scriptlet: iptables-nft-1.8.8-3.amzn2023.0.2.x86_64 8/10
Installing : libcgroup-3.0-1.amzn2023.0.1.x86_64 9/10
Running scriptlet: docker-25.0.8-1.amzn2023.0.1.x86_64 10/10
Installing : docker-25.0.8-1.amzn2023.0.1.x86_64 10/10
Running scriptlet: docker-25.0.8-1.amzn2023.0.1.x86_64 10/10
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.
Verifying : containerd-1.7.27-1.amzn2023.0.1.x86_64 1/10
Verifying : docker-25.0.8-1.amzn2023.0.1.x86_64 2/10
Verifying : iptables-libs-1.8.8-3.amzn2023.0.2.x86_64 3/10
Verifying : iptables-nft-1.8.8-3.amzn2023.0.2.x86_64 4/10
Verifying : libcgroup-3.0-1.amzn2023.0.1.x86_64 5/10
Verifying : libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64 6/10
Verifying : libnfnetlink-1.0.1-19.amzn2023.0.2.x86_64 7/10
Verifying : libnftnl-1.2.2-2.amzn2023.0.2.x86_64 8/10
Verifying : pigz-2.5-1.amzn2023.0.3.x86_64 9/10
Verifying : runc-1.2.4-1.amzn2023.0.1.x86_64 10/10
Installed:
containerd-1.7.27-1.amzn2023.0.1.x86_64 docker-25.0.8-1.amzn2023.0.1.x86_64
iptables-libs-1.8.8-3.amzn2023.0.2.x86_64 iptables-nft-1.8.8-3.amzn2023.0.2.x86_64
libcgroup-3.0-1.amzn2023.0.1.x86_64 libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64
libnfnetlink-1.0.1-19.amzn2023.0.2.x86_64 libnftnl-1.2.2-2.amzn2023.0.2.x86_64
i-0df1e7ca096769f3a (sepmsql)
PublicIPs: 43.204.148.115 PrivateIPs: 172.31.6.231
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

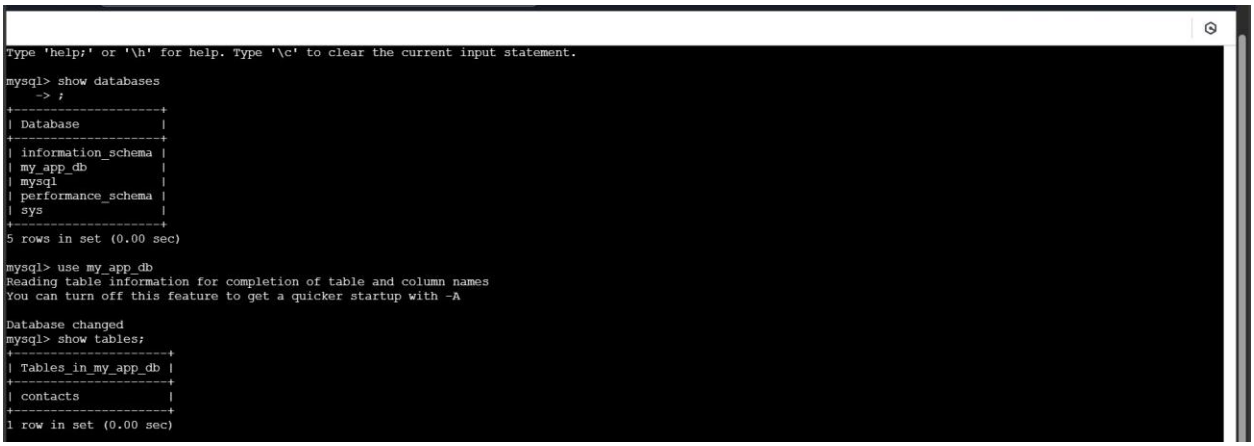

Search
[Alt+S]
Asia Pacific (Mumbai)
Ashish Gupta
Installed:
containerd-1.7.27-1.amzn2023.0.1.x86_64 docker-25.0.8-1.amzn2023.0.1.x86_64
iptables-libs-1.8.8-3.amzn2023.0.2.x86_64 iptables-nft-1.8.8-3.amzn2023.0.2.x86_64
libcgroup-3.0-1.amzn2023.0.1.x86_64 libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64
libnfnetlink-1.0.1-19.amzn2023.0.2.x86_64 libnftnl-1.2.2-2.amzn2023.0.2.x86_64
pigz-2.5-1.amzn2023.0.3.x86_64 runc-1.2.4-1.amzn2023.0.1.x86_64
Complete!
[ec2-user@ip-172-31-6-231 ~]$ sudo service docker start
Redirecting to /bin/systemctl start docker.service
[ec2-user@ip-172-31-6-231 ~]$ sudo systemctl start docker
[ec2-user@ip-172-31-6-231 ~]$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; preset: disabled)
Active: active (running) since Wed 2025-04-02 05:19:13 UTC; 2min 15s ago
TriggeredBy: ● docker.socket
Docs: https://docs.docker.com
Process: 27201 ExecStartPre=/bin/mkdir -p /run/docker (code=exited, status=0/SUCCESS)
Process: 27203 ExecStartPre=/usr/libexec/docker/docker-setup-runtimes.sh (code=exited, status=0/SUC>
Main PID: 27204 (dockerd)
Tasks: 7
Memory: 28.3M
CPU: 270ms
CGroup: /system.slice/docker.service
└─27204 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock --default-u>
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal systemd[1]: Starting docker.service - Docke>
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.2>
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.3>
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.7>
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.7>
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.7>
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.8>
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal systemd[1]: Started docker.service - Docker>
lines 1-22/22 (END)
[ec2-user@ip-172-31-6-231 ~]$ sudo docker pull philippaul/node-mysql-app:02
02: Pulling from philippaul/node-mysql-app
2ff1d7c41c74: Pull complete
b253aeafeaa7: Pull complete
3d2201bd995c: Pull complete
1de76e268b10: Pull complete
d9a8df589451: Pull complete
i-0df1e7ca096769f3a (sepmsql)
PublicIPs: 43.204.148.115 PrivateIPs: 172.31.6.231
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Tasks: 7
Memory: 28.3M
CPU: 270ms
CGroup: /system.slice/docker.service
└─27204 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock --default-
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal systemd[1]: Starting docker.service - Docke
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.2
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.2
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.7
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.7
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.7
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal dockerd[27204]: time="2025-04-02T05:19:13.8
Apr 02 05:19:13 ip-172-31-6-231.ap-south-1.compute.internal systemd[1]: Started docker.service - Docke
lines 1-22/22 (END)
[ec2-user@ip-172-31-6-231 ~]$ sudo docker pull philippaul/node-mysql-app:02
02: Pulling from philippaul/node-mysql-app
2ff1d7c41c74: Pull complete
b253aeafeaa7: Pull complete
9d2201bd995c: Pull complete
1de76e268b10: Pull complete
d9a8df589451: Pull complete
6f51ee005dea: Pull complete
5f32ed3c3f27: Pull complete
0c8cc2f24a4d: Pull complete
0d27a8e86132: Pull complete
b35ca9a95db0: Pull complete
46a182df3db1: Pull complete
f5b1a7ebee97: Pull complete
ff7978b844b1: Pull complete
Digest: sha256:f7c1cffb42a2f4a40b626b0d03f8b83bbc8ef3f88d0682cd43f395bf9e42966b
Status: Downloaded newer image for philippaul/node-mysql-app:02
docker.io/philippaul/node-mysql-app:02
[ec2-user@ip-172-31-6-231 ~]$ sudo docker imags
docker: 'imags' is not a docker command.
See 'docker --help'
[ec2-user@ip-172-31-6-231 ~]$ sudo docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
philippaul/node-mysql-app 02 4b941beb4207 5 months ago 923MB
[ec2-user@ip-172-31-6-231 ~]$ sudo docker run --rm -p 80:3000 -e DB_HOST="sepmdb.c96e2iow4kyp.ap-south-1.rds.amazonaws.com" -e DB_USER="admin" -e DB_PASSWORD="ashishgupta" -d philippaul/node-mysql-app:02
062754ecd70b96bd8b98b656539fd8d8b0b33277e7f78015a977b230ca4d17e8
[ec2-user@ip-172-31-6-231 ~]$
i-0df1e7ca096769f3a (sepmsql)
PublicIPs: 43.204.148.115 PrivateIPs: 172.31.6.231
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


Contact App
ashish@gmail.com
Add Username
Show All Contacts
ashish@gmail.com
Delete
MPrashant Academy


Contact App
Enter Username
Add Username
Show All Contacts
ashish@gmail.com
Delete
ashishgupta@gmail.com
Delete
Delete
MPrashant Academy


aws
Search
[Alt+S]
Asia Pacific (Mumbai)
adityadikonda
etch the logs of a container
ec2-user@ip-172-30-0-203 ~]$ sudo docker run -it --rm mysql:8.0 mysql -h t1224.c3aaqi8w4qsq.ap-south-1.rds.amazonaws.com -u admin -p
nable to find image 'mysql:8.0' locally
.0: Pulling from library/mysql
ea172a6e83b: Pull complete
8e01aa53f13: Pull complete
5fa3211d7a7: Pull complete
53b8441f7e6: Pull complete
1339a14fa1a: Pull complete
e386ff914e3: Pull complete
3272c957f26: Pull complete
106a4902288: Pull complete
036f4325df2d: Pull complete
d34979e7120: Pull complete
de67a2f637e5: Pull complete
igest: sha256:bf577825b52ab281d6281fb281eabbfdc73507eda8f2c2745790251533ef0306
tatus: Downloaded newer image for mysql:8.0
nter password:
elcome to the MySQL monitor. Commands end with ; or \g.
our MySQL connection id is 28
erver version: 8.0.40 Source distribution
opyright (c) 2000, 2025, Oracle and/or its affiliates.
racle is a registered trademark of Oracle Corporation and/or its
ffiliates. Other names may be trademarks of their respective
wners.
ype 'help;' or '\h' for help. Type '\c' to clear the current input statement.
ysql> show databases
-> ;
+--------------------+
| Database |
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> show databases
-> ;
+--------------------+
| Database |
+--------------------+
| information_schema |
| my_app_db |
| mysql |
| performance_schema |
| sys |
+--------------------+
5 rows in set (0.00 sec)
mysql> use my_app_db
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Database changed
mysql> show tables;
+----------------------+
| Tables_in_my_app_db |
+----------------------+
| contacts |
+----------------------+
1 row in set (0.00 sec)

Conclusion:
Docker revolutionizes the software development and deployment process by providing a powerful platform for containerization. By encapsulating applications and their dependencies into lightweight, portable containers,