Project Documentation – Employee Management Web Application (AWS Deployment)

1. Project Overview

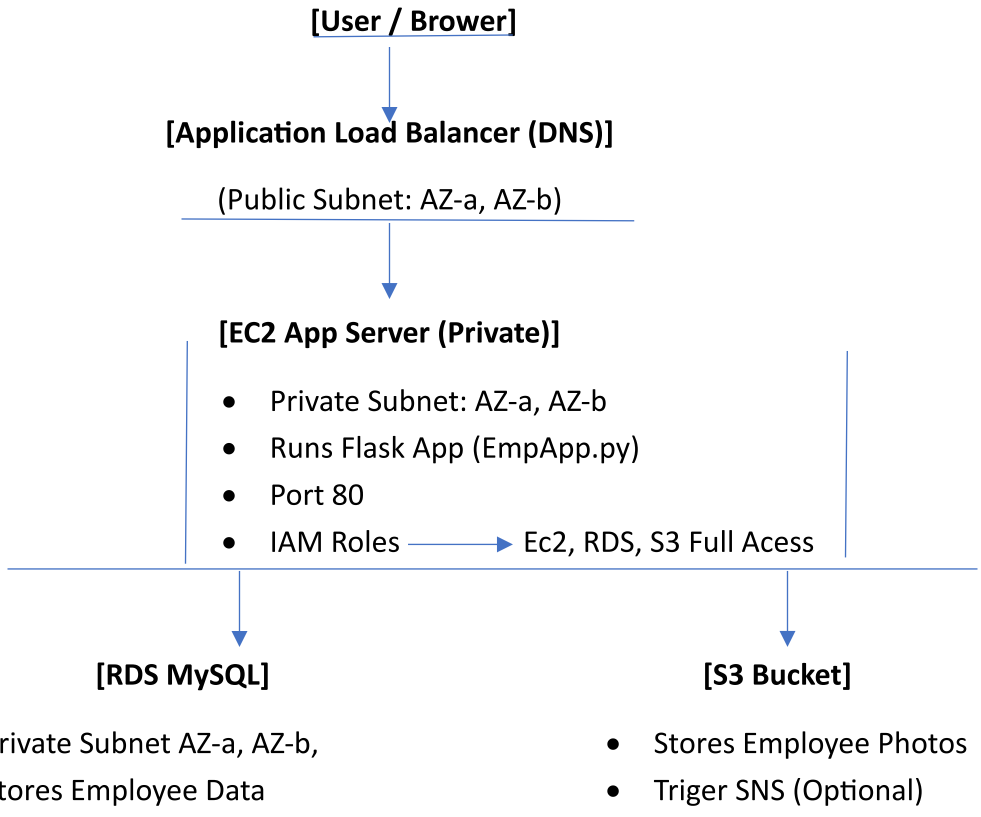
This Project is an Employee Management Web Application Built using Python (Flask) and deployed on AWS Cloud using:

- EC2 (Basion host & App Server)
- Application Load Balancer (Public Access)
- RDS MySQL Database
- S3 Bucket (Employee Photo Storage)
- IAM Roles + Security Groups
- VPC with Public & Private Subnet

The Application allows users to:

- Add Employee Details
- Uploads Employee Photos
- View Employee Infromation

2. Architecture Diagram (Text Format)



[User / Brower]

[Application Load Balancer (DNS)]

(Public Subnet: AZ-a, AZ-b)

[EC2 App Server (Private)]

- Private Subnet: AZ-a, AZ-b
- Runs Flask App (EmpApp.py)
- Port 80
- IAM Roles → Ec2, RDS, S3 Full Acess

[RDS MySQL]

- Private Subnet AZ-a, AZ-b,
- Stores Employee Data

[S3 Bucket]

- Stores Employee Photos
- Triger SNS (Optional)

3. AWS Service Used

EC2 Instance

- Hosts Python Flask application
- Runs EmpApp.py on port 80
- IAM Role attached for S3 access

Application Load Balancer (ALB)

- Public facing
- Handles HTTP traffic (port 80)
- Routes to target group → EC2

RDS MySQL Database

- Stores employee details
- Private subnet (not publicly accessible)

S3 Bucket

- Stores uploaded employee images

IAM Roles

- EC2 →S3 full ascess

Security Groups

- ALB SG → allow all traffics HTTP 80 from anywhere
- EC2, S3 → allow traffics
- RDS SG → allow MySQL 3306 from EC2 SG

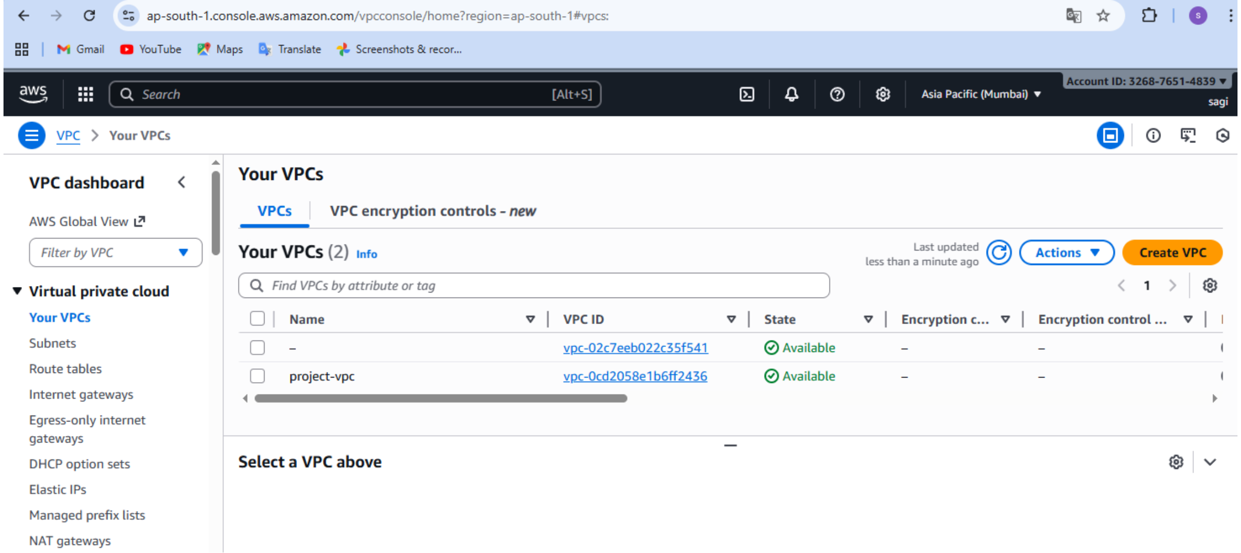
VPC Components

- Public subnets for ALB
- Private subnet for EC2 & RDS
- NAT Gateway for private subnet internet access
- Internet Gateway for public subnet internet access
- Route tables

4. **Step-by-Step Deployment:**

Step 1: Create VPC & Networking

➔ Create VPC (project-vpc) - 10.0.0.0/16



Your VPCs

Your VPCs (2) Info

	Name	VPC ID	State	Encryption c...	Encryption control ...
□	–	vpc-02c7eeb022c35f541	✓ Available	–	–
□	project-vpc	vpc-0cd2058e1b6ff2436	✓ Available	–	–

➔ Create 2 Public Subnets (2 public subnet because of load balancer is using(ALB))

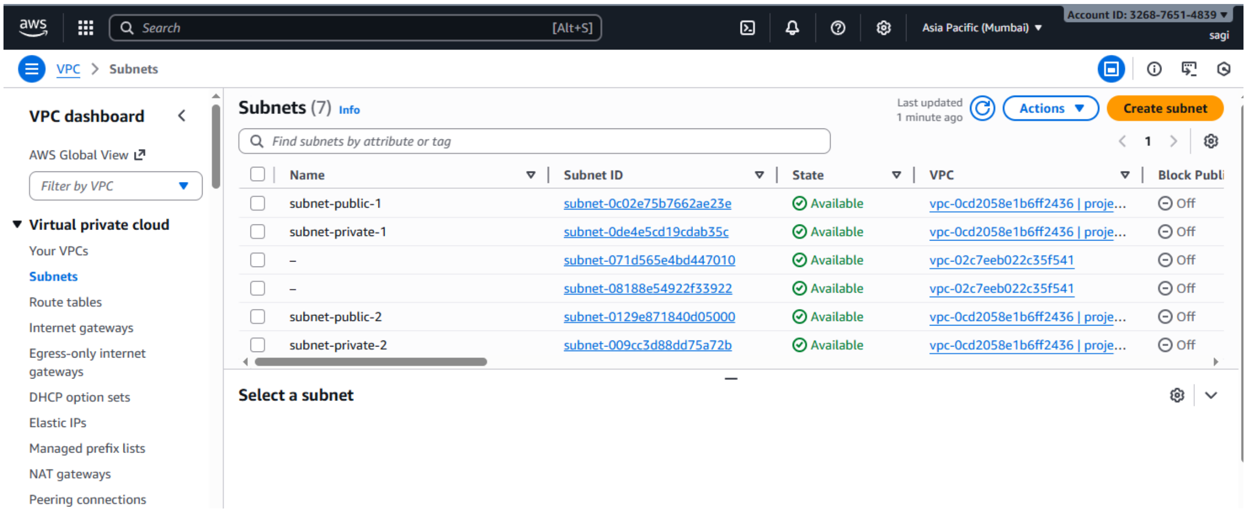
➔ Create 2 Private Subnets (for EC2, RDS)

➔ subnet1 - 10.20.1.0/24 - public-1 (az-ap-south-1a)

➔ subnet2 - 10.20.2.0/24 - private-1 (az-ap-south-1a)

➔ Subnet3 - 10.20.3.0/24 - public-2 (az-ap-south-1b) (2 public subnet because of load balancer is using)

➔ Subnet4 - 10.20.4.0/24 - private-2 (az-ap-south-1b)

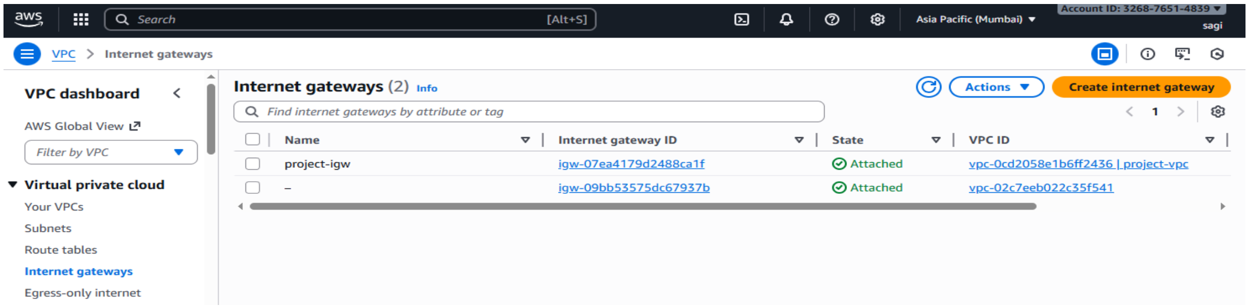


Subnets (7) Info

	Name	Subnet ID	State	VPC	Block Publi
□	subnet-public-1	subnet-0c02e75b7662ae23e	✓ Available	vpc-0cd2058e1b6ff2436 | proje...	⊖ Off
□	subnet-private-1	subnet-0de4e5cd19cdab35c	✓ Available	vpc-0cd2058e1b6ff2436 | proje...	⊖ Off
□	–	subnet-071d565e4bd447010	✓ Available	vpc-02c7eeb022c35f541	⊖ Off
□	–	subnet-08188e54922f33922	✓ Available	vpc-02c7eeb022c35f541	⊖ Off
□	subnet-public-2	subnet-0129e871840d05000	✓ Available	vpc-0cd2058e1b6ff2436 | proje...	⊖ Off
□	subnet-private-2	subnet-009cc3d88dd75a72b	✓ Available	vpc-0cd2058e1b6ff2436 | proje...	⊖ Off

➔ Create IGW, NAT-gateway, Route table

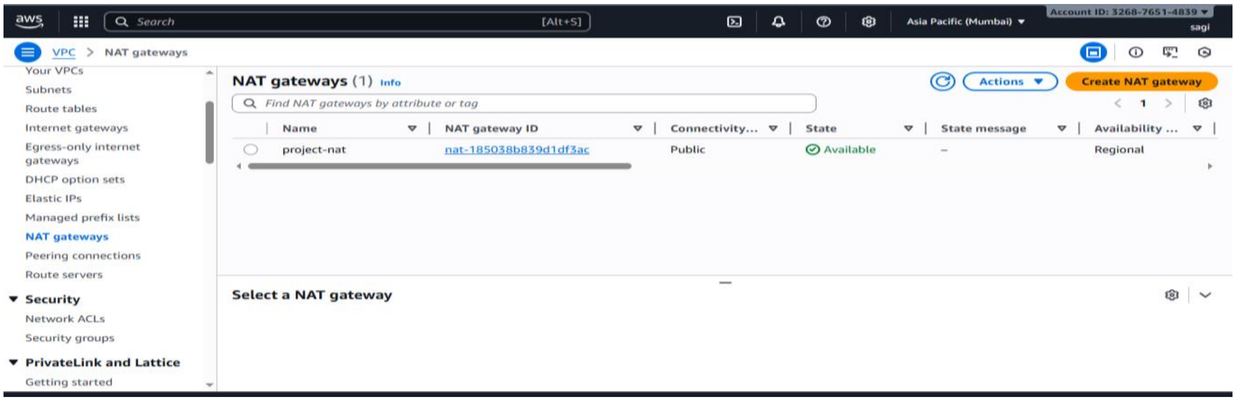
- Create Internet Gateway attach to VPC -- project-IGW



Internet gateways (2) Info

Name	Internet gateway ID	State	VPC ID
project-igw	igw-07ea4179d2488ca1f	Attached	vpc-0cd2058e1b6ff2436 | project-vpc
–	igw-09bb53575dc67937b	Attached	vpc-02c7eeb022c35f541

- Create NAT-gateway -- project-NAT -- subnet (public-1) -- Allocate elastic ip.

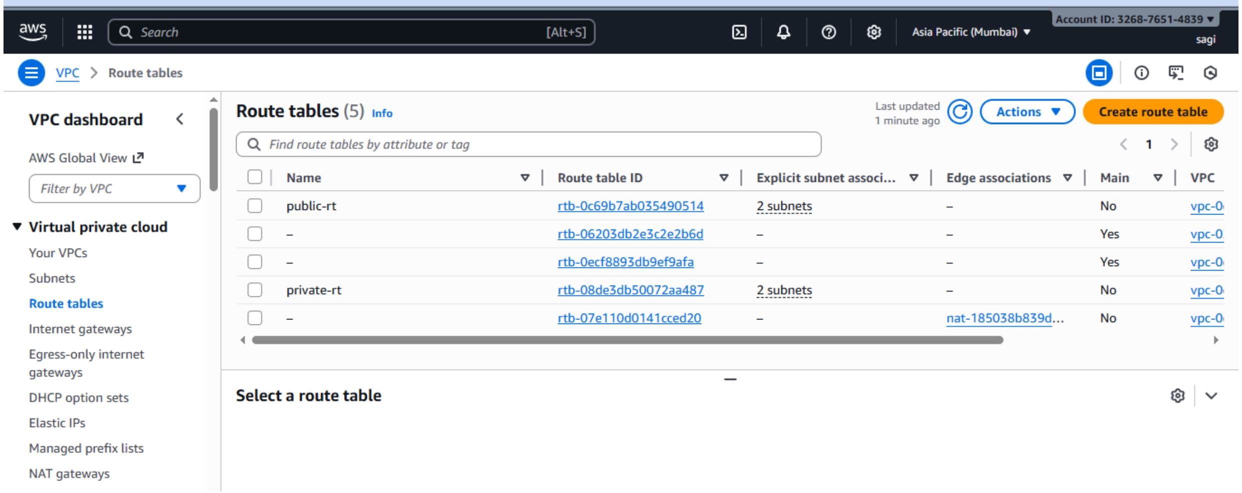


NAT gateways (1) Info

Name	NAT gateway ID	Connectivity...	State	State message	Availability ...
project-nat	nat-185038b839d1df3ac	Public	Available	–	Regional

Select a NAT gateway

- Create Route table -- public-rt -- project-vpc. Select public-rt -- edit route -- add route 0.0.0.0/0. -- target (created - IGW). Subnet associations -- edit -- select public1 & public2
- Create Route table -- private-rt -- project-vpc. Select private-rt -- edit route -- add route 0.0.0.0/0. -- target (created - nat) . Subnet associations -- edit -- select private-1 & private-2



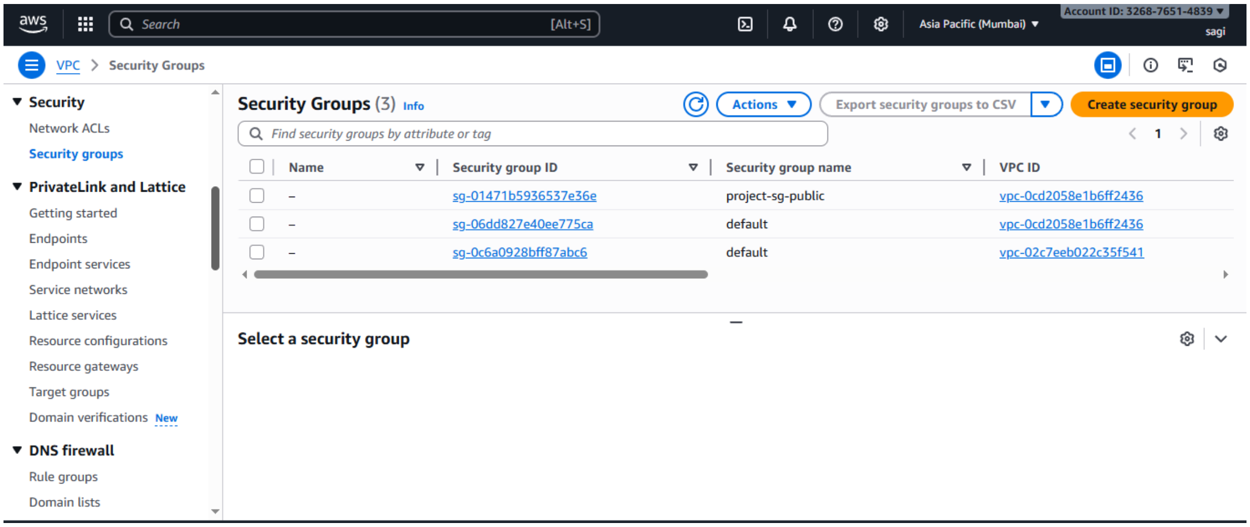
Route tables (5) Info

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
public-rt	rtb-0c69b7ab035490514	2 subnets	–	No	vpc-0
–	rtb-06203db2e3c2e2b6d	–	–	Yes	vpc-0
–	rtb-0ecf8893db9ef9afa	–	–	Yes	vpc-0
private-rt	rtb-08de3db50072aa487	2 subnets	–	No	vpc-0
–	rtb-07e110d0141cced20	–	nat-185038b839d...	No	vpc-0

Select a route table

Step 2: Create Security Group

- ➔ Create security group -- project-sg -- project-vpc -- inbound rule - all traffic – anyway
- ➔ Check the Network ACLs -- inbound rule (100 allow) & outbound rule (100 allow) + subnet association (all subnet available).



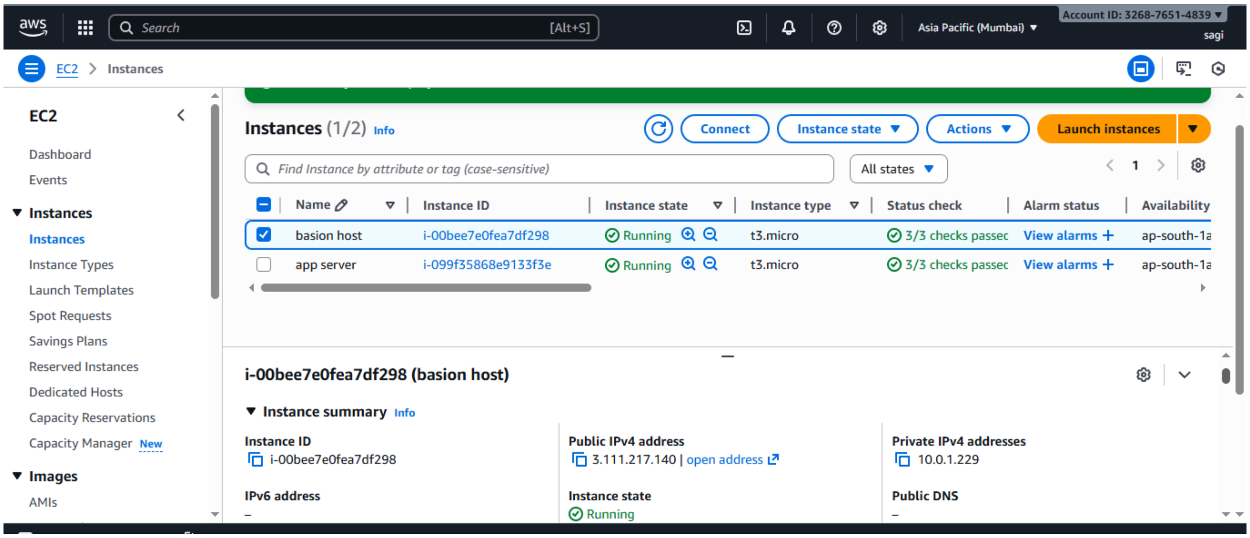
Security Groups (3) Info

☐	Name	Security group ID	Security group name	VPC ID
☐	–	sg-01471b5936537e36e	project-sg-public	vpc-0cd2058e1b6ff2436
☐	–	sg-06dd827e40ee775ca	default	vpc-0cd2058e1b6ff2436
☐	–	sg-0c6a0928bff87abc6	default	vpc-02c7eeb022c35f541

Select a security group

Step 3: Launch EC2 instance:

- ➔ Select Ubuntu AMI for both EC2
- ➔ Create Bastion Host (public-1 subnet) -- project-vpc -- auto-assign IP (enable) -- projects. (SG=SSH & all traffic allow)



Instances (1/2) Info

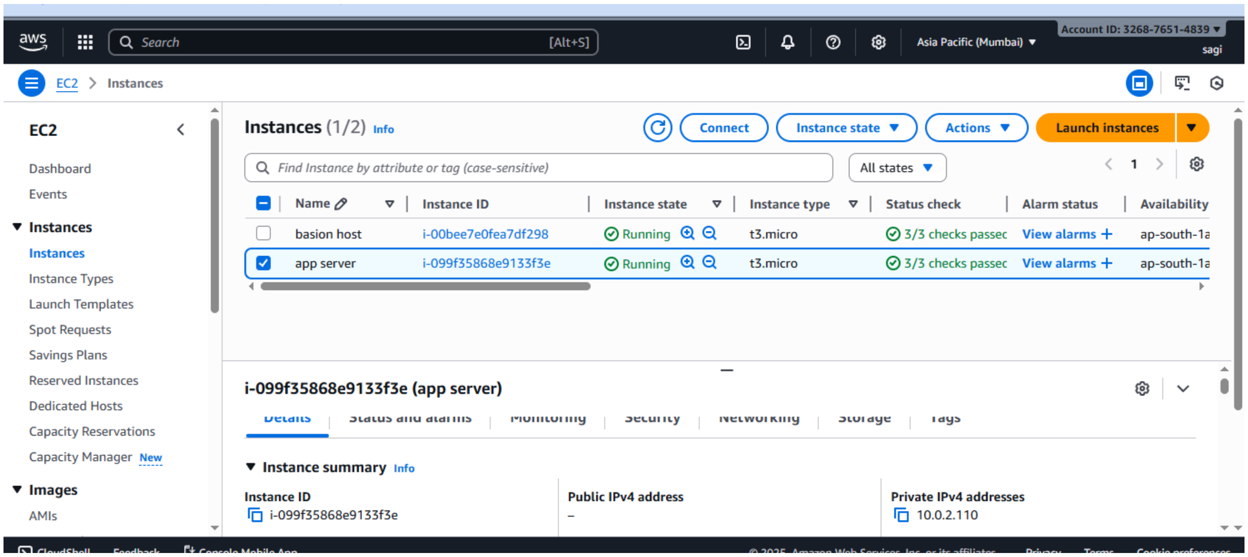
☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
☑	basion host	i-00bee7e0fea7df298	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	ap-south-1a
☐	app server	i-099f35868e9133f3e	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	ap-south-1a

i-00bee7e0fea7df298 (basion host)

▼ Instance summary Info

Instance ID	Public IPv4 address	Private IPv4 addresses
i-00bee7e0fea7df298	3.111.217.140 | open address	10.0.1.229
IPv6 address	**Instance state**	**Public DNS**
–	✓ Running	–

➔ Create Application-server (private-1 subnet) -- project-vpc -- auto-assign IP (disable) -- projects. ((SG=SSH & all traffic allow))



Instances (1/2) Info

☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
☐	basion host	i-00bee7e0fea7df298	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	ap-south-1a
☑	app server	i-099f35868e9133f3e	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	ap-south-1a

i-099f35868e9133f3e (app server)

▼ Instance summary Info

Instance ID	Public IPv4 address	Private IPv4 addresses
i-099f35868e9133f3e	–	10.0.2.110

➔ Step 4: login to Bastion-host server & run following commands

```
# sudo apt-get update
```

Copy key pair form your PC to Bastion-host PC because we are using Application server for security level.

Terminal = `scp -i keypair.pem keypair.pem ubuntu@public ip:/home/ubuntu/`

`#chmod 400 keypair-name` For security purpose

Login in application-server → `#ssh -I keypair-name.pem ubuntu@private ip.`

```
#apt-get update -y
```

Git hub link to clone the application code : https://github.com/zubair3337/aws-project-1.git

```
#sudo apt-git install git (if required)
# git init
#git clone "https://github.com/zubair3337/aws-project-1.git"
#ls
#cd aws-project-1 (check files)
#apt-get install apache2 -y
#mv aws-project-1/ /var/www/html/
#cd /var/www/html/ (cross check all files & Directory should be available)
#apt-get install mysql-client
```

➔ For Sql-client

```
#sudo apt-get install mysql-client
```

- ➔ For python and related frameworks

  #sudo apt-get install python3-pip

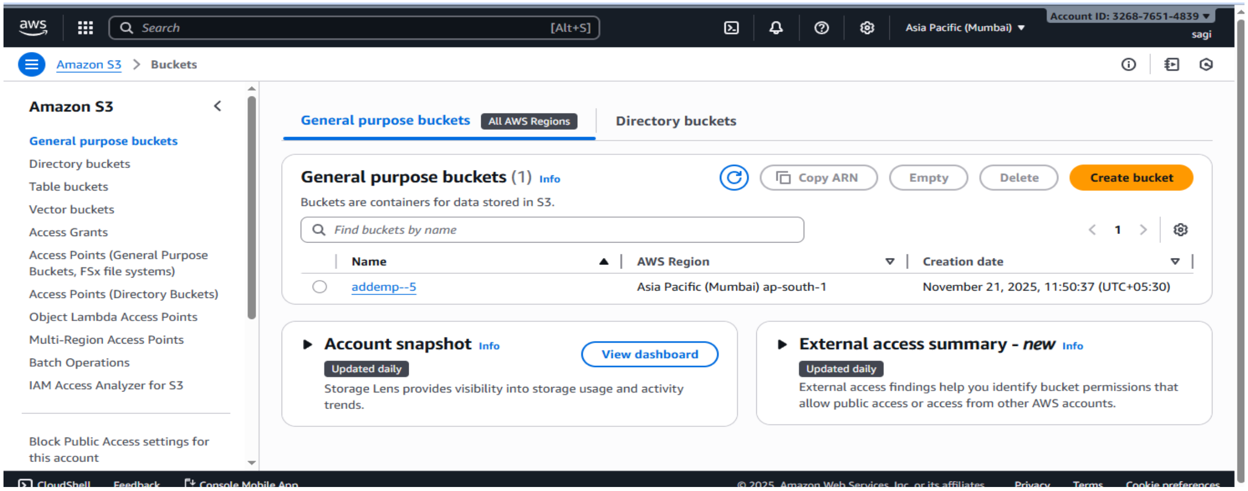
  #sudo apt-get install python3-flask

  #sudo apt-get install python3-pymysql

  #sudo apt-get install python3-boto3

Step 5: Create S3 bucket (to store image)

- ➔ In Amazon S3, Create S3 bucket
- ➔ Bucket name: addemp--5 (allow public access)



General purpose buckets (1) Info

Buckets are containers for data stored in S3.

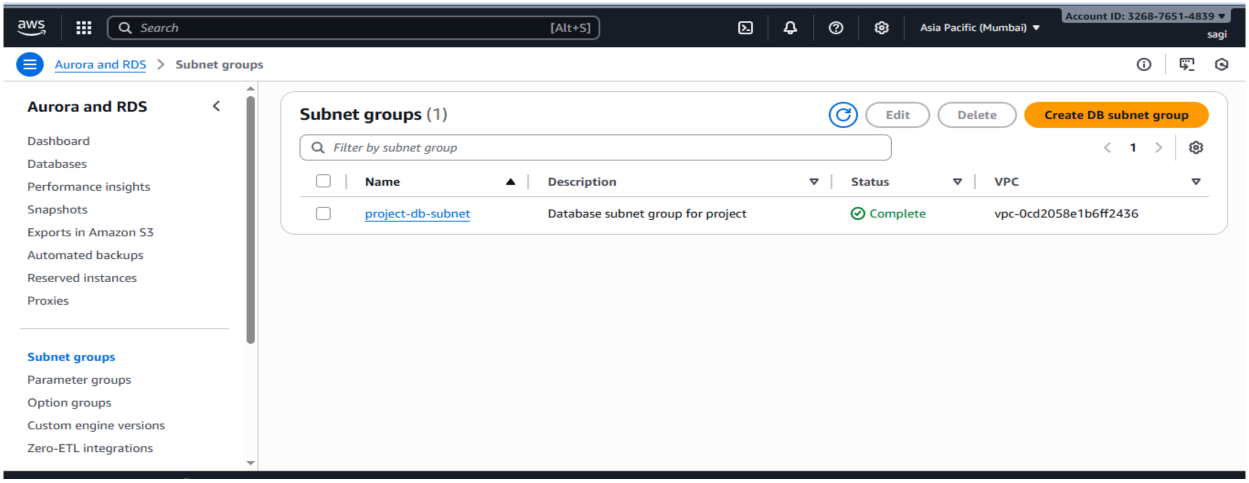
Name	AWS Region	Creation date
○ addemp--5	Asia Pacific (Mumbai) ap-south-1	November 21, 2025, 11:50:37 (UTC+05:30)

Storage Lens provides visibility into storage usage and activity trends.

External access findings help you identify bucket permissions that allow public access or access from other AWS accounts.

Step 6: Create RDS instance (to store text data)

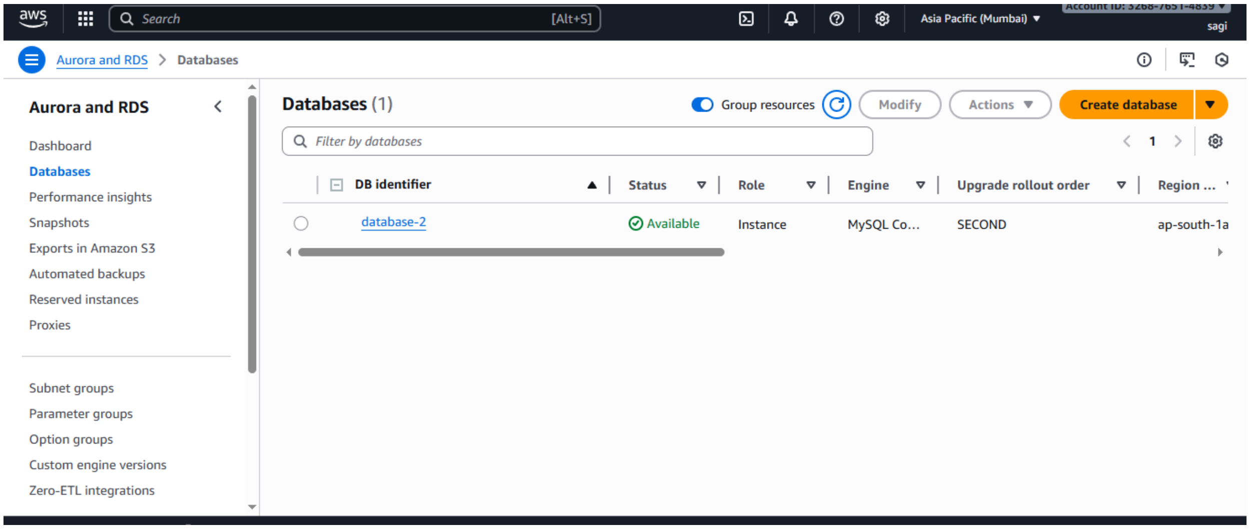
- ➔ Before creating RDS instance - in RDS dashboard go to "subnet group" and create DB subnet group.
- ➔ subnet group Name : project - custom vpc - AZ-2a & AZ-2b
- ➔ subnet : 10.0.4.0/24 & 10.0.2.0/24)



Subnet groups (1)

Name	Description	Status	VPC
□ project-db-subnet	Database subnet group for project	✓ Complete	vpc-0cd2058e1b6ff2436

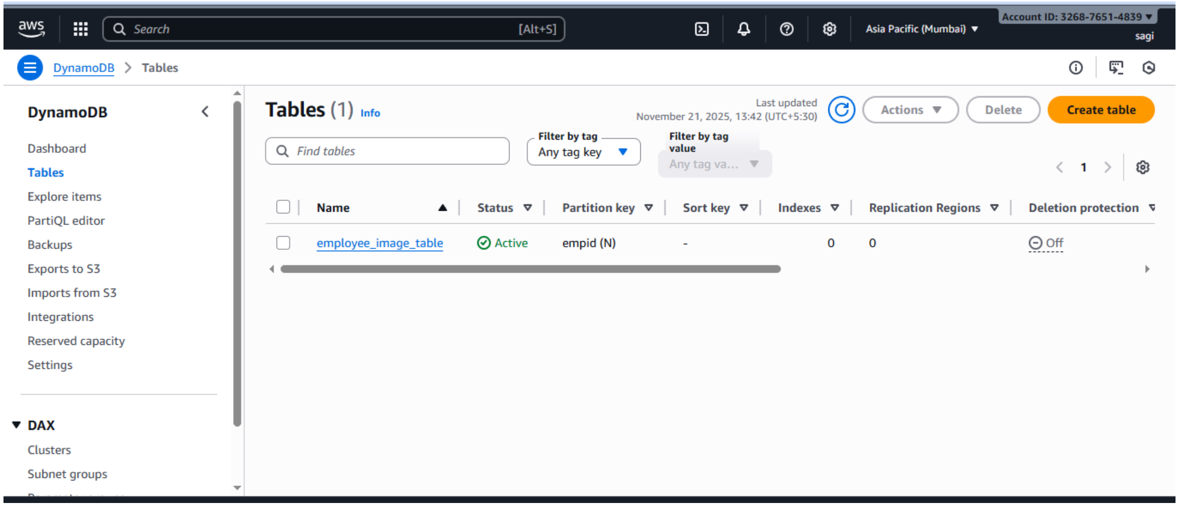
- ➔ Now Create RDS - Standard -- mysql -- free tier -- admin -- admin123 -- custom VPC -- subnet group "project" -- don't want public access -
- ➔ SG - project-sG -- (AZ us-east-2a)
- ➔ Additional configuration : database-name (employee) -- disable auto backup -- no monitoring -- no upgrades.



DB identifier	Status	Role	Engine	Upgrade rollout order	Region ...
database-2	Available	Instance	MySQL Co...	SECOND	ap-south-1a

Step 7: Create DynamoDB

- ➔ Table name : employee_image_table
- ➔ Primary key : empid -- number



Last updated
November 21, 2025, 13:42 (UTC+5:30)

Name	Status	Partition key	Sort key	Indexes	Replication Regions	Deletion protection
employee_image_table	Active	empid (N)	-	0	0	Off

Step 8: login to application-server (/var/www/html) and add some changes in aws-project-1 directory

```
#vim config.py
```

```
Customhouse = "RDS endpoint name"

Customiser = "admin"

Custom pass = "admin123"

Custom db = "employee"

Custom bucket = "addemp--5"

Custom region = "ap-south-1"

Customisable = "employee_image_table"
```

➔ :wq for Save and Exit

```
#vim EmpApp.py
```

Save image file metadata in DynamoDB (after this line add some changes)

Region_name='ap-south-1" (find region and change with current region)

Table name : 'employee_image_table'

➔ Save & exit

Step 9: login to RDS mysql server to create table use following command

Syntax : `mysql -h "RDS end point name" -u admin -p`

➔ Asking for password and connect with MySql >

```
#SHOW DATABASES;
```

We can see "employee" database is created.

```
#use employee;
```

Database changed

```
#CREATE TABLE employee (emp_id VARCHAR(20), first_name VARCHAR(20), last_name VARCHAR(20), primary_skills VARCHAR(20), location VARCHAR(20));
```

Query ok, 0 row affected

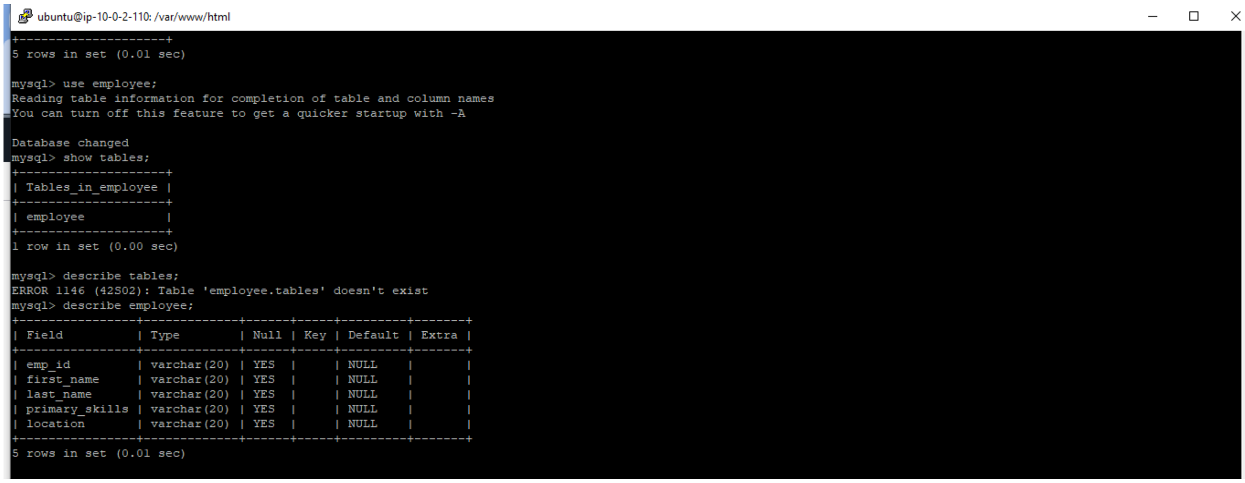
#show tables;

employee

#describe employee;

Kindly check all table field are available or not

#exit (from mysql) Bye



ubuntu@ip-10-0-2-110: /var/www/html

```
+--------------------+
5 rows in set (0.01 sec)

mysql> use employee;
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_employee |
+--------------------+
| employee           |
+--------------------+
1 row in set (0.00 sec)

mysql> describe tables;
ERROR 1146 (42S02): Table 'employee.tables' doesn't exist
mysql> describe employee;
+----------------+-------------+------+-----+---------+-------+
| Field          | Type        | Null | Key | Default | Extra |
+----------------+-------------+------+-----+---------+-------+
| emp_id         | varchar(20) | YES  |     | NULL    |       |
| first_name     | varchar(20) | YES  |     | NULL    |       |
| last_name      | varchar(20) | YES  |     | NULL    |       |
| primary_skills | varchar(20) | YES  |     | NULL    |       |
| location       | varchar(20) | YES  |     | NULL    |       |
+----------------+-------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

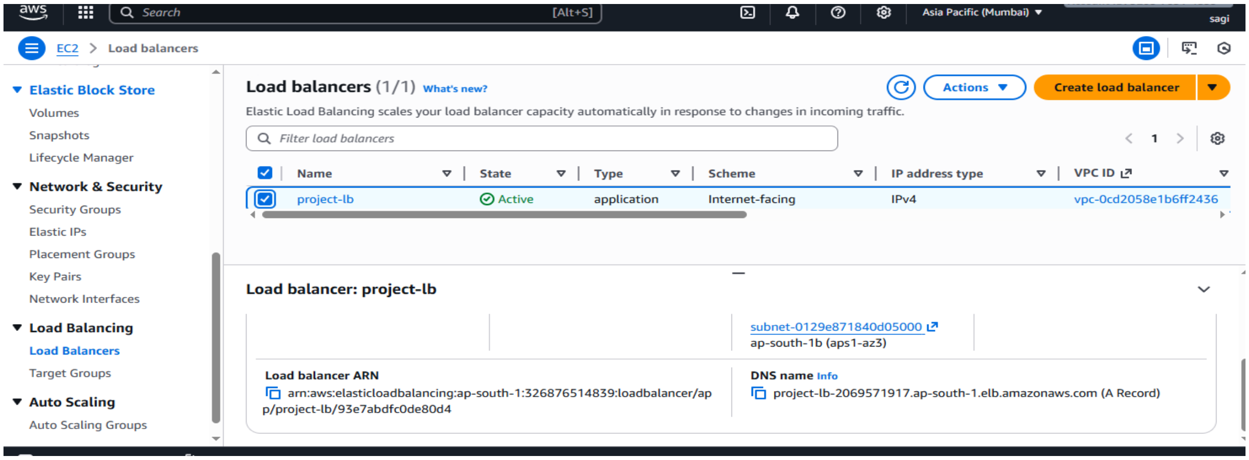
#sudo systemctl stop apache2 (need to stop apache2 because python & apache2 both running on port 80)

#sudo python3 EmpApp.py (to run application)

Application is running on private instance so need to add load balancer for accessing application

Step 9: Create Target group and Application Load Balancer

➔ Create target group (select application server) Load Balancer (Application load balancer) should be in public subnet (AZ 1a & 1b) & Target load balancer



Load balancers (1/1)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

☑ Name	State	Type	Scheme	IP address type	VPC ID
☑ project-lb	✓ Active	application	Internet-facing	IPv4	vpc-0cd2058e1b6ff2436

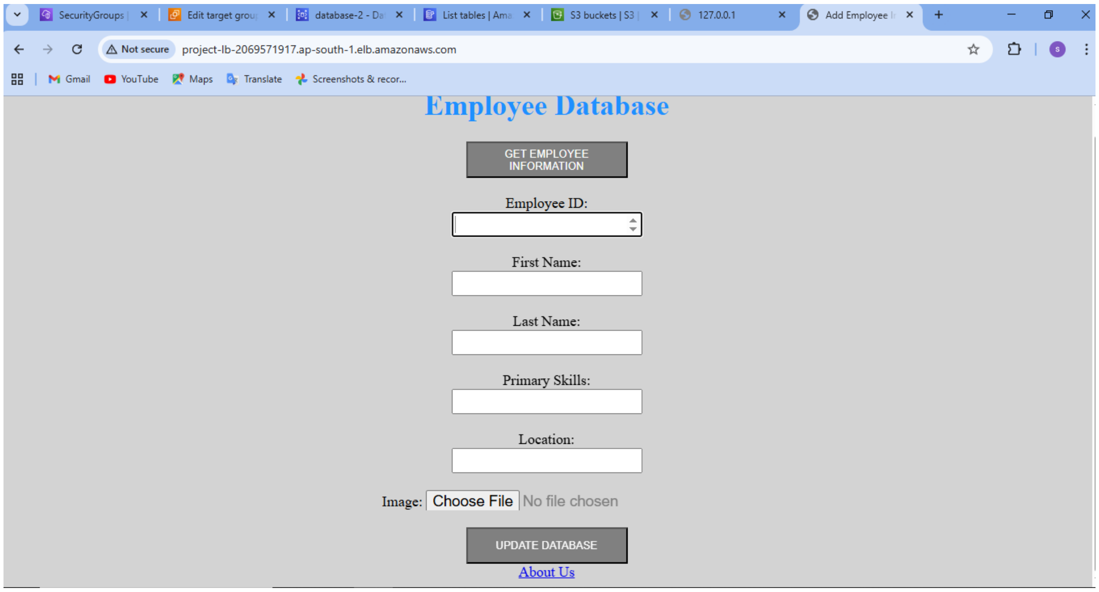
Load balancer: project-lb

subnet-0129e871840d05000
ap-south-1b (aps1-az3)

Load balancer ARN
arn:aws:elasticloadbalancing:ap-south-1:326876514839:loadbalancer/app/project-lb/93e7abdfc0de80d4

DNS name Info
project-lb-2069571917.ap-south-1.elb.amazonaws.com (A Record)

- Copy ALB DNS record and paste on browser you will able to see web application.



project-lb-2069571917.ap-south-1.elb.amazonaws.com

Employee Database

Employee ID:

First Name:

Last Name:

Primary Skills:

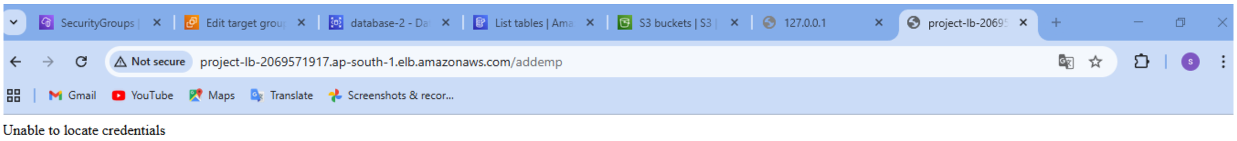
Location:

Image: No file chosen

- Now you can insert data in web application but "update database" not working. So for that need to attach IAM role

Step 10: Create IAM role and attached with application-server

- IAM -- roles -- create role -- aWs services -- ec2 -- permission -- administrator access & EC2 full access, RDS access -- role name (project-role) -- create role
- Select Application server -- Action -- security -- modify IAM role -- role name(project-role) – apply
- Now if you click on update database button it'll work.
- While click on go back button it will not work showing connection refuse.

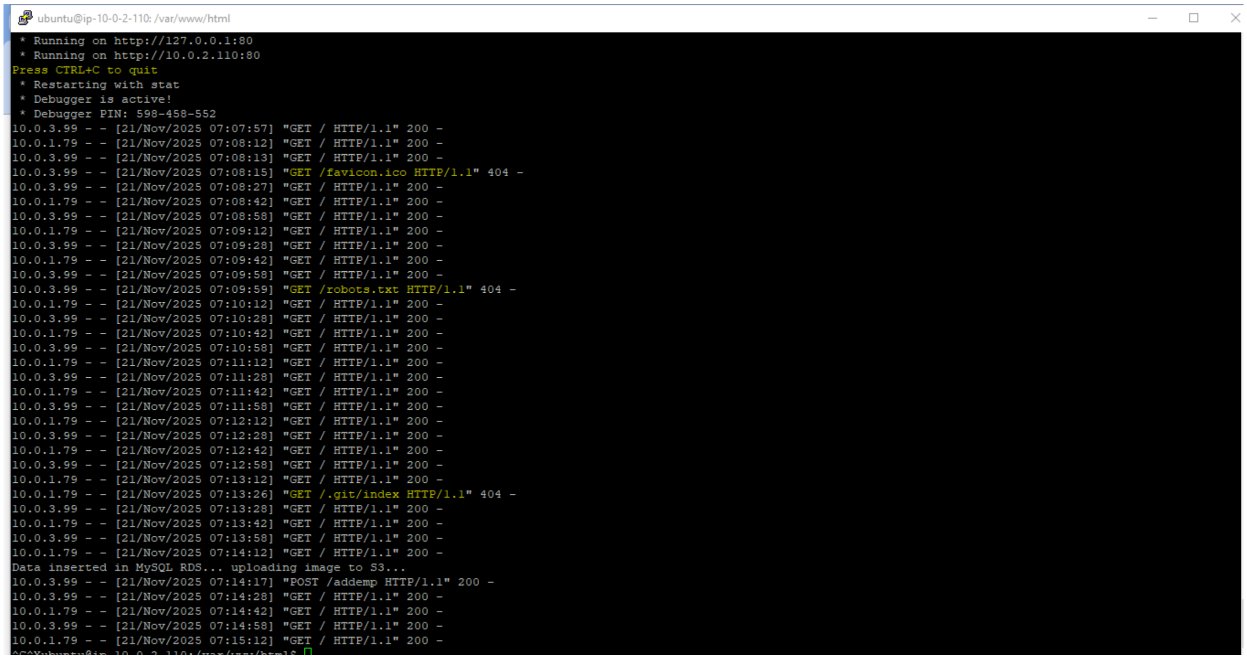


project-lb-2069571917.ap-south-1.elb.amazonaws.com/addemp

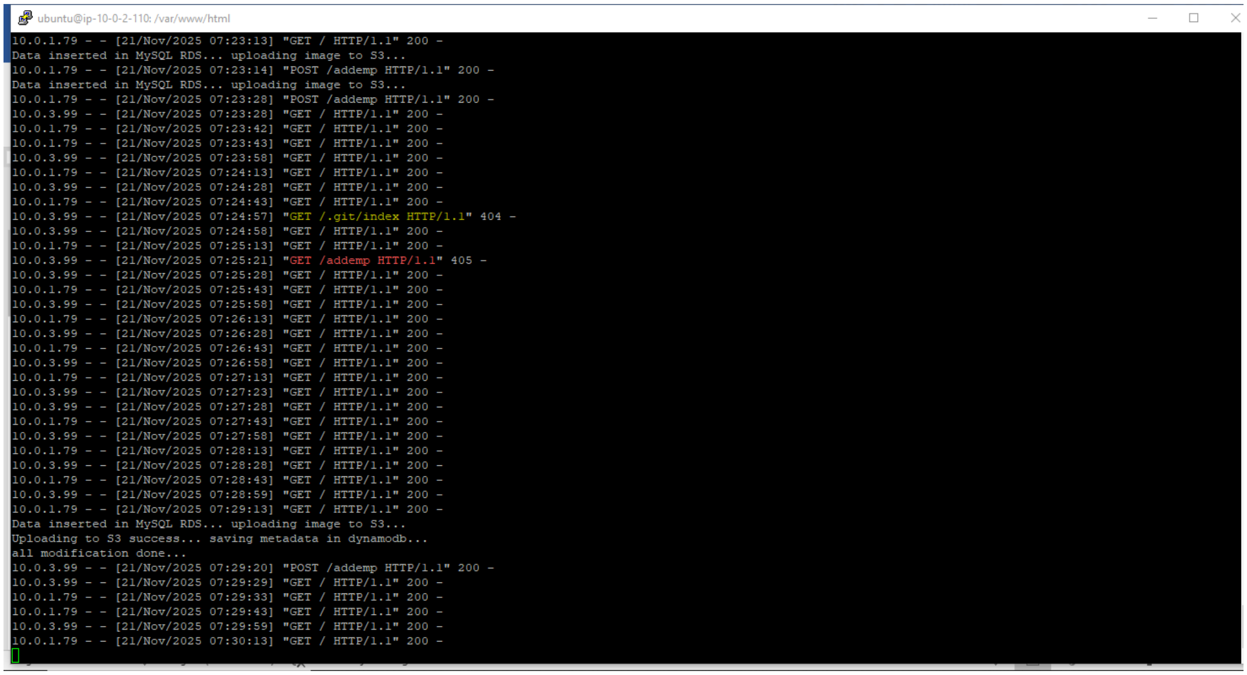
Unable to locate credentials

Step 11: Login to Application Server and Run Script/Application

# sudo python EmpApp.py (Do not stop this command, Create duplicate session)



```
 * Running on http://127.0.0.1:80
 * Running on http://10.0.2.110:80
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 598-458-552
10.0.3.99 - - [21/Nov/2025 07:07:57] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:08:12] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:08:13] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:08:15] "GET /favicon.ico HTTP/1.1" 404 -
10.0.3.99 - - [21/Nov/2025 07:08:27] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:08:42] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:08:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:09:12] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:09:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:09:42] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:09:58] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:09:59] "GET /robots.txt HTTP/1.1" 404 -
10.0.1.79 - - [21/Nov/2025 07:10:12] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:10:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:10:42] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:10:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:11:12] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:11:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:11:42] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:11:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:12:12] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:12:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:12:42] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:12:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:13:12] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:13:26] "GET /.git/index HTTP/1.1" 404 -
10.0.3.99 - - [21/Nov/2025 07:13:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:13:42] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:13:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:14:12] "GET / HTTP/1.1" 200 -
Data inserted in MySQL RDS... uploading image to S3...
10.0.3.99 - - [21/Nov/2025 07:14:17] "POST /addemp HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:14:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:14:42] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:14:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:15:12] "GET / HTTP/1.1" 200 -
^C^Xubuntu@ip-10-0-2-110:/var/www/html$
```



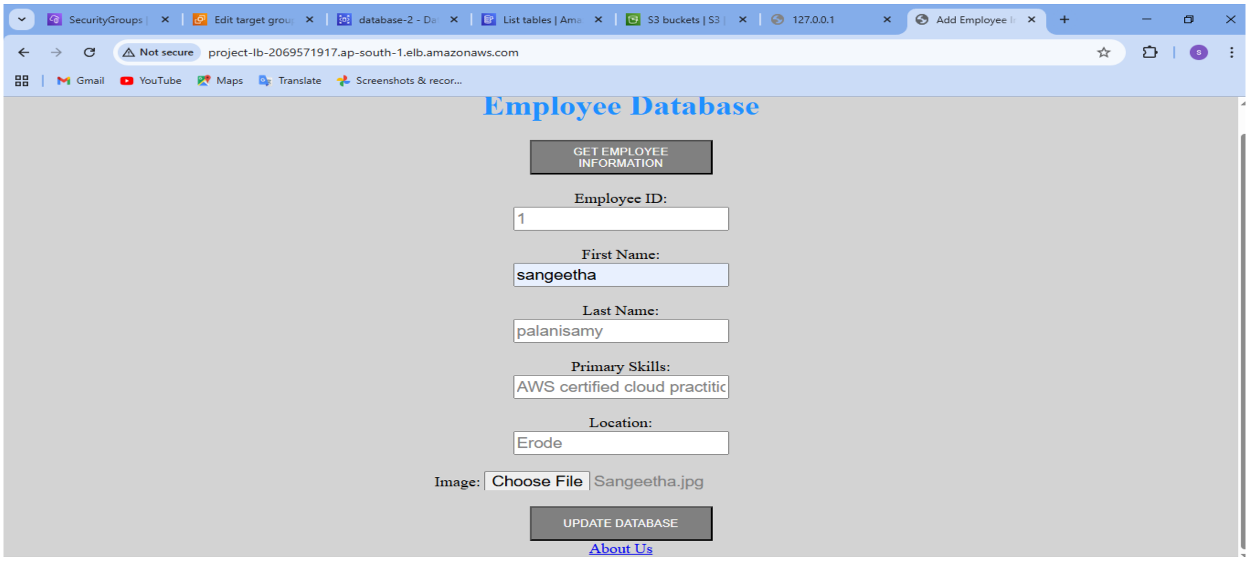
```
10.0.1.79 - - [21/Nov/2025 07:23:13] "GET / HTTP/1.1" 200 -
Data inserted in MySQL RDS... uploading image to S3...
10.0.1.79 - - [21/Nov/2025 07:23:14] "POST /addemp HTTP/1.1" 200 -
Data inserted in MySQL RDS... uploading image to S3...
10.0.1.79 - - [21/Nov/2025 07:23:28] "POST /addemp HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:23:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:23:42] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:23:43] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:23:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:24:13] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:24:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:24:43] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:24:57] "GET /.git/index HTTP/1.1" 404 -
10.0.3.99 - - [21/Nov/2025 07:24:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:25:13] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:25:21] "GET /addemp HTTP/1.1" 405 -
10.0.3.99 - - [21/Nov/2025 07:25:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:25:43] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:25:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:26:13] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:26:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:26:43] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:26:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:27:13] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:27:23] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:27:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:27:43] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:27:58] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:28:13] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:28:28] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:28:43] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:28:59] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:29:13] "GET / HTTP/1.1" 200 -
Data inserted in MySQL RDS... uploading image to S3...
Uploading to S3 success... saving metadata in dynamodb...
all modification done...
10.0.3.99 - - [21/Nov/2025 07:29:20] "POST /addemp HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:29:29] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:29:33] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:29:43] "GET / HTTP/1.1" 200 -
10.0.3.99 - - [21/Nov/2025 07:29:59] "GET / HTTP/1.1" 200 -
10.0.1.79 - - [21/Nov/2025 07:30:13] "GET / HTTP/1.1" 200 -
```

*Note : don't stop python3 EmpApp.py (if its stop website will stop working, to perform other task open new terminal windows and complete extra task)

Step 12: go to /var/www/html/aws-project-1/templates/

#sudo vim AddEmpOutput.html (and go to "action" option and change ip address with Load Balancer endpoint with "" commas).
#sudo vim GetEmpOutput.html (go to "button" line and after "formaction" just delete http://ipaddress)

- ➔ Now in portal if back button press it will work.
- ➔ Update Employee Database



project-lb-2069571917.ap-south-1.elb.amazonaws.com

Employee Database

GET EMPLOYEE INFORMATION

Employee ID: 1

First Name: sangeetha

Last Name: palanisamy

Primary Skills: AWS certified cloud practitic

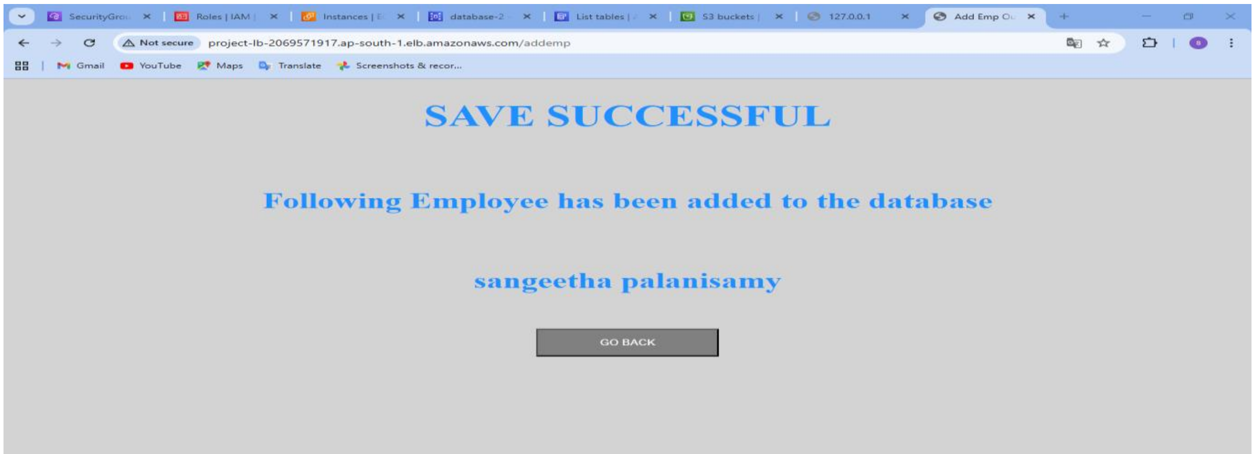
Location: Erode

Image: Choose File Sangeetha.jpg

UPDATE DATABASE

About Us

- ➔ Successfully data saved.



project-lb-2069571917.ap-south-1.elb.amazonaws.com/addemp

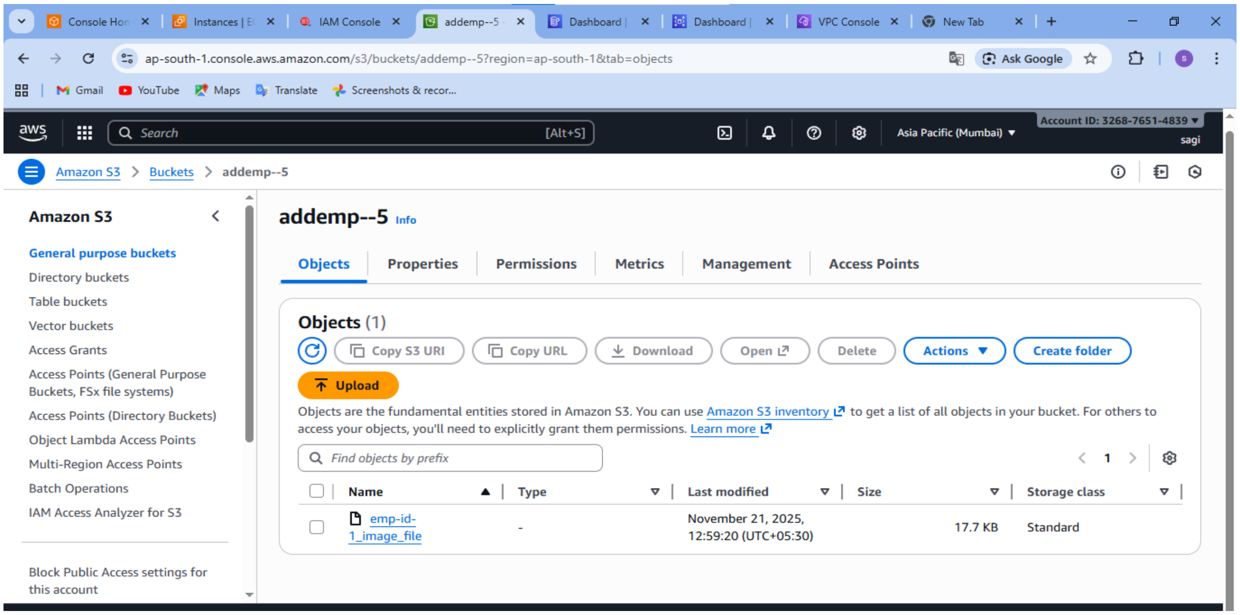
SAVE SUCCESSFUL

Following Employee has been added to the database

sangeetha palanisamy

GO BACK

- ➔ Now database will update in portal/Application.
- ➔ Open S3 & Dynamodb so some data will available which is upload by user.
- ➔ Updated Empolyee photo in S3

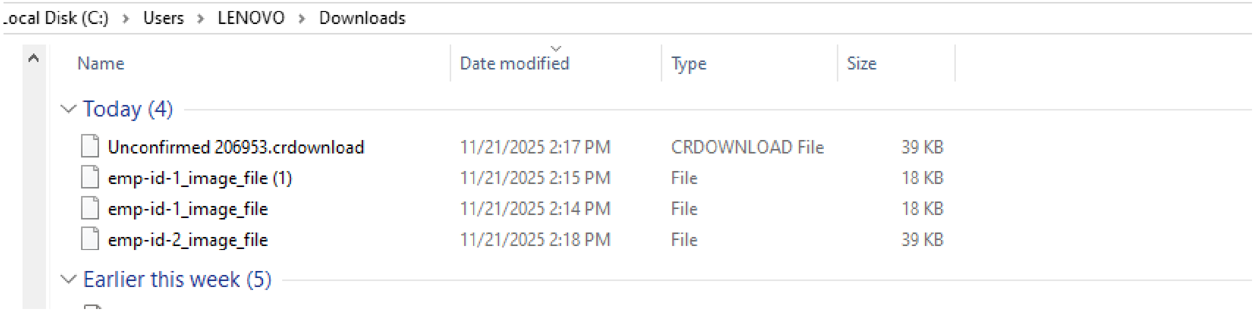


addemp--5 Info

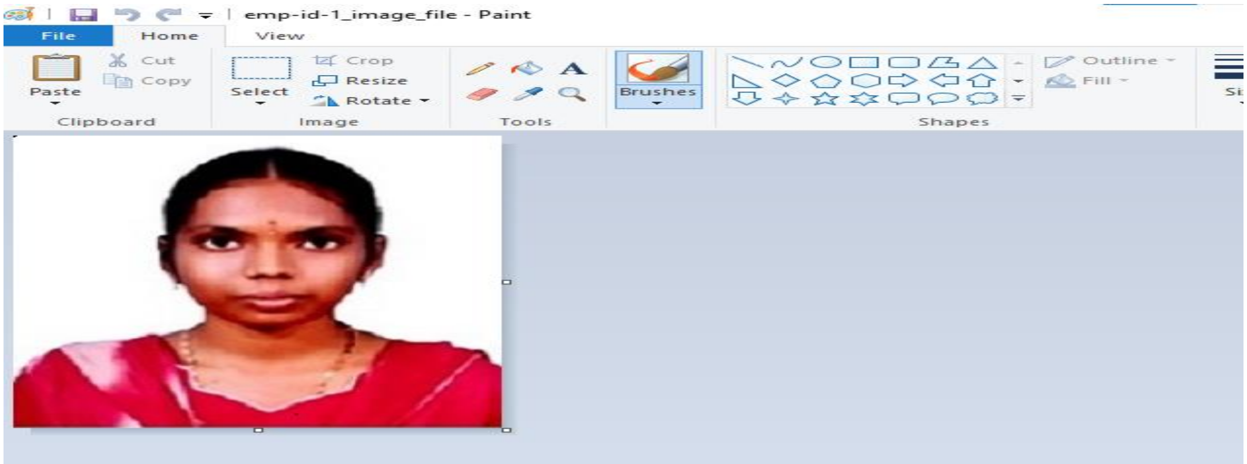
Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

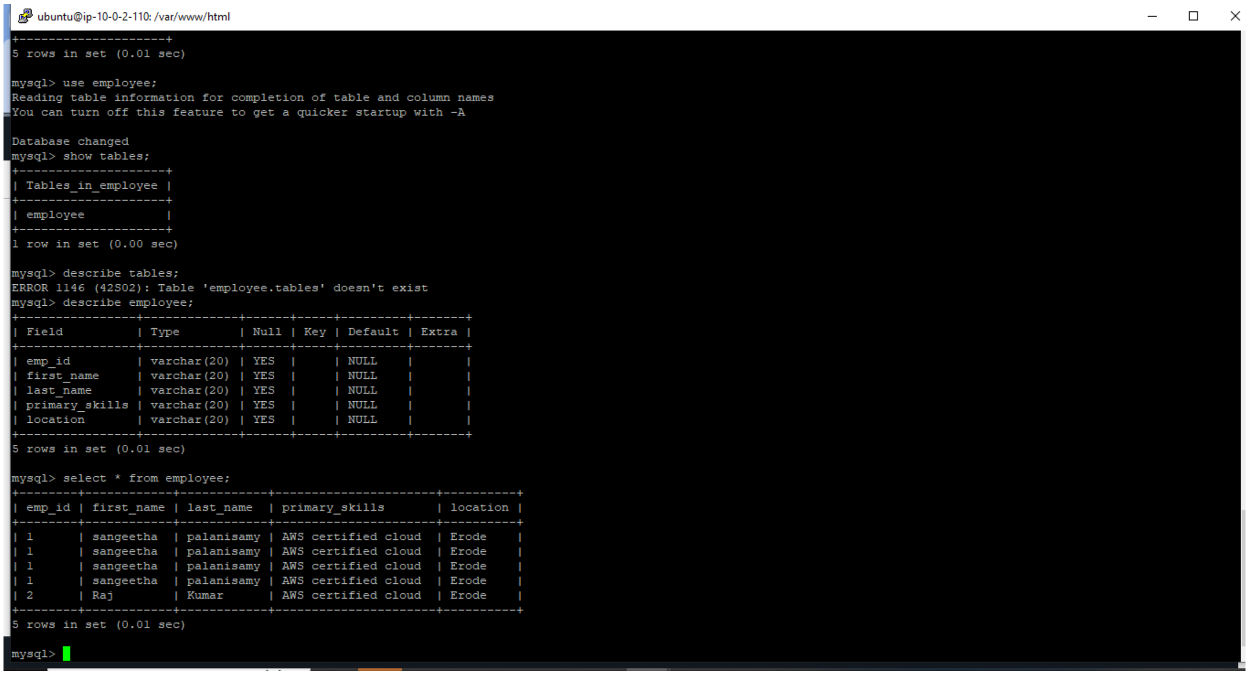
	Name	Type	Last modified	Size	Storage class
□	emp-id-1_image_file	-	November 21, 2025, 12:59:20 (UTC+05:30)	17.7 KB	Standard


.ocal Disk (C:) › Users › LENOVO › Downloads

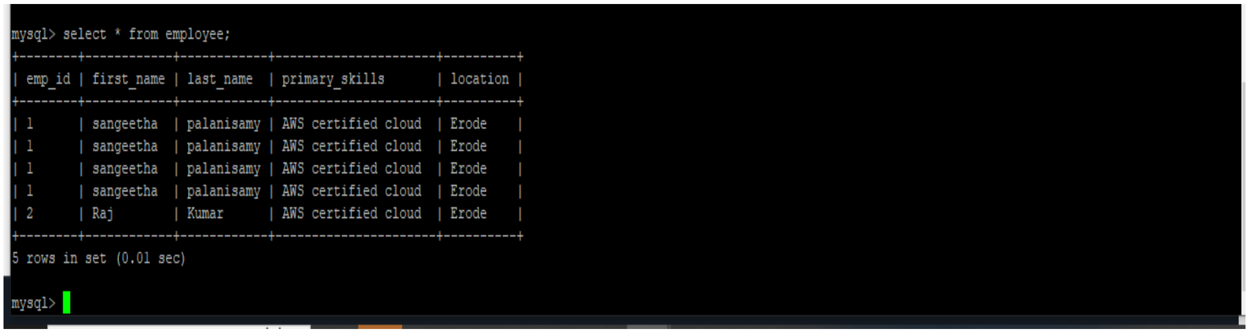
Name	Date modified	Type	Size
Today (4)			
Unconfirmed 206953.crdownload	11/21/2025 2:17 PM	CRDOWNLOAD File	39 KB
emp-id-1_image_file (1)	11/21/2025 2:15 PM	File	18 KB
emp-id-1_image_file	11/21/2025 2:14 PM	File	18 KB
emp-id-2_image_file	11/21/2025 2:18 PM	File	39 KB
Earlier this week (5)			



- Saved employee datas:



```
+--------------------+
5 rows in set (0.01 sec)

mysql> use employee;
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_employee |
+--------------------+
| employee           |
+--------------------+
1 row in set (0.00 sec)

mysql> describe tables;
ERROR 1146 (42S02): Table 'employee.tables' doesn't exist
mysql> describe employee;
+----------------+-------------+------+-----+---------+-------+
| Field          | Type        | Null | Key | Default | Extra |
+----------------+-------------+------+-----+---------+-------+
| emp_id         | varchar(20) | YES  |     | NULL    |       |
| first_name     | varchar(20) | YES  |     | NULL    |       |
| last_name      | varchar(20) | YES  |     | NULL    |       |
| primary_skills | varchar(20) | YES  |     | NULL    |       |
| location       | varchar(20) | YES  |     | NULL    |       |
+----------------+-------------+------+-----+---------+-------+
5 rows in set (0.01 sec)

mysql> select * from employee;
+--------+------------+------------+----------------------+----------+
| emp_id | first_name | last_name  | primary_skills       | location |
+--------+------------+------------+----------------------+----------+
| 1      | sangeetha  | palanisamy | AWS certified cloud  | Erode    |
| 1      | sangeetha  | palanisamy | AWS certified cloud  | Erode    |
| 1      | sangeetha  | palanisamy | AWS certified cloud  | Erode    |
| 1      | sangeetha  | palanisamy | AWS certified cloud  | Erode    |
| 2      | Raj        | Kumar      | AWS certified cloud  | Erode    |
+--------+------------+------------+----------------------+----------+
5 rows in set (0.01 sec)

mysql> 
```



```
mysql> select * from employee;
+--------+------------+------------+----------------------+----------+
| emp_id | first_name | last_name  | primary_skills       | location |
+--------+------------+------------+----------------------+----------+
| 1      | sangeetha  | palanisamy | AWS certified cloud  | Erode    |
| 1      | sangeetha  | palanisamy | AWS certified cloud  | Erode    |
| 1      | sangeetha  | palanisamy | AWS certified cloud  | Erode    |
| 1      | sangeetha  | palanisamy | AWS certified cloud  | Erode    |
| 2      | Raj        | Kumar      | AWS certified cloud  | Erode    |
+--------+------------+------------+----------------------+----------+
5 rows in set (0.01 sec)

mysql> 
```

================================Optional Steps==================================

Step 13: SNS Created for message/email notification.

- SNS Dashboard : create topic --- standard --- topic name (event) --- rest thing as default and create topic.
- Now create subscription : topic name select (event) --- protocol (email) -- email id (any).
- Logan to your email and click on email link and confirm subscription.

➔ Go to snstopic --- open(even)---edit---access policy (just delete last two lines "stringEquals - was:source....." And add following lines.

```
"ArnLike": {
  "aws:SourceArn": "arn:aws:s3:*:*:<bucket name>"
}
```

➔ Save

➔ Select created s3 bucket and go to properties --- Event Notification --- create event notification --- event name (any) --- select "all object create events" --- select SNS topic name (event) and save

➔ So after this settings you will get notification if some 1 is adding any detail in portal.

======== If you have domain ready use following option otherwise no problem ========

Step 14: Route 53

➔ As we discuss kindly create free domain with "freenom" website and copy the domain name and create new hosted zone in route 53.

➔ Create hosted zone : domain name (your created domain) ---- public hosted zone --- create.

➔ Change 4 "NS" record with freedom "name server" record.

Click on create record
Record name : www ---- record type : A record
Route traffic : select created load balancer
Enable "Alias" option.
Routing policy : simple routing

➔ Create.

➔ After some minutes kindly type your freedom domain name in browser it will work.

Done with small project