

Deploying End To End website on AWS

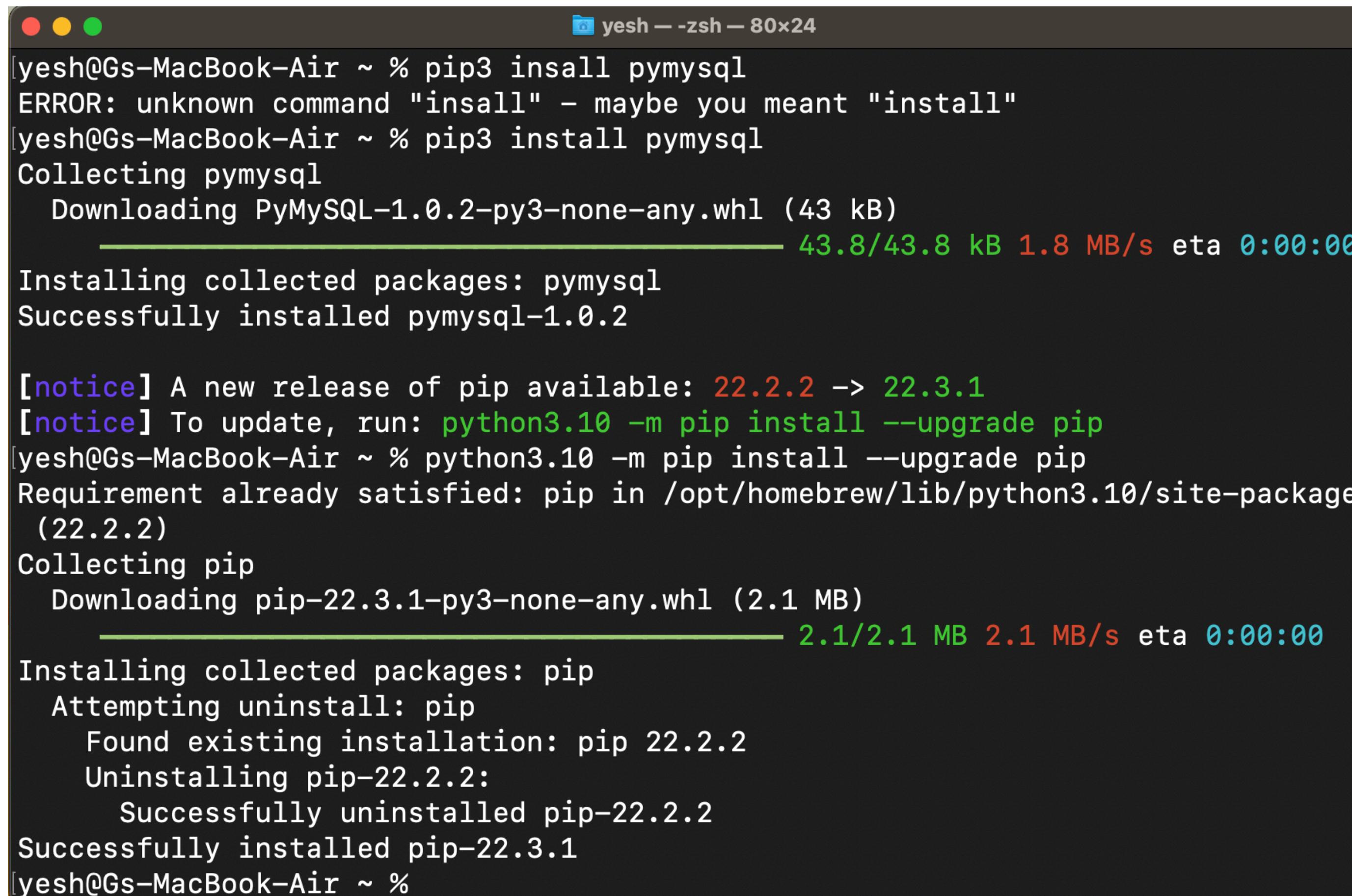


-G.YESHWANTH GOUD (HU21CSC1O200001)
BSC.CSCS

STEP 1

INSTALL PYMYSQL ON YOUR TERMINAL

- ENTER THIS COMMAND “`pip3 install pymysql` ” AND PRESS ENTER



```
yesh@Gs-MacBook-Air ~ % pip3 insall pymysql
ERROR: unknown command "insall" – maybe you meant "install"
yesh@Gs-MacBook-Air ~ % pip3 install pymysql
Collecting pymysql
  Downloading PyMySQL-1.0.2-py3-none-any.whl (43 kB)
    ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 43.8/43.8 kB 1.8 MB/s eta 0:00:00
Installing collected packages: pymysql
Successfully installed pymysql-1.0.2

[notice] A new release of pip available: 22.2.2 -> 22.3.1
[notice] To update, run: python3.10 -m pip install --upgrade pip
yesh@Gs-MacBook-Air ~ % python3.10 -m pip install --upgrade pip
Requirement already satisfied: pip in /opt/homebrew/lib/python3.10/site-packages
(22.2.2)
Collecting pip
  Downloading pip-22.3.1-py3-none-any.whl (2.1 MB)
    ━━━━━━━━━━━━━━━━━━━━━━━ 2.1/2.1 kB 2.1 MB/s eta 0:00:00
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 22.2.2
    Uninstalling pip-22.2.2:
      Successfully uninstalled pip-22.2.2
  Successfully installed pip-22.3.1
yesh@Gs-MacBook-Air ~ %
```

STEP 3

CREATE AN RDS ON AWS

- KEEP MYSQL DATABASE NAME AS “employee”.

The screenshot shows the AWS RDS console interface for the 'employee' database. The top navigation bar includes links for Console Home, EC2, RDS, S3, IAM, CloudFront, Elastic Beanstalk, AWS Cost Explorer, and a user dropdown for 'yeshwanth'. The main page displays the 'Summary' and 'Connectivity & security' tabs for the 'employee' database.

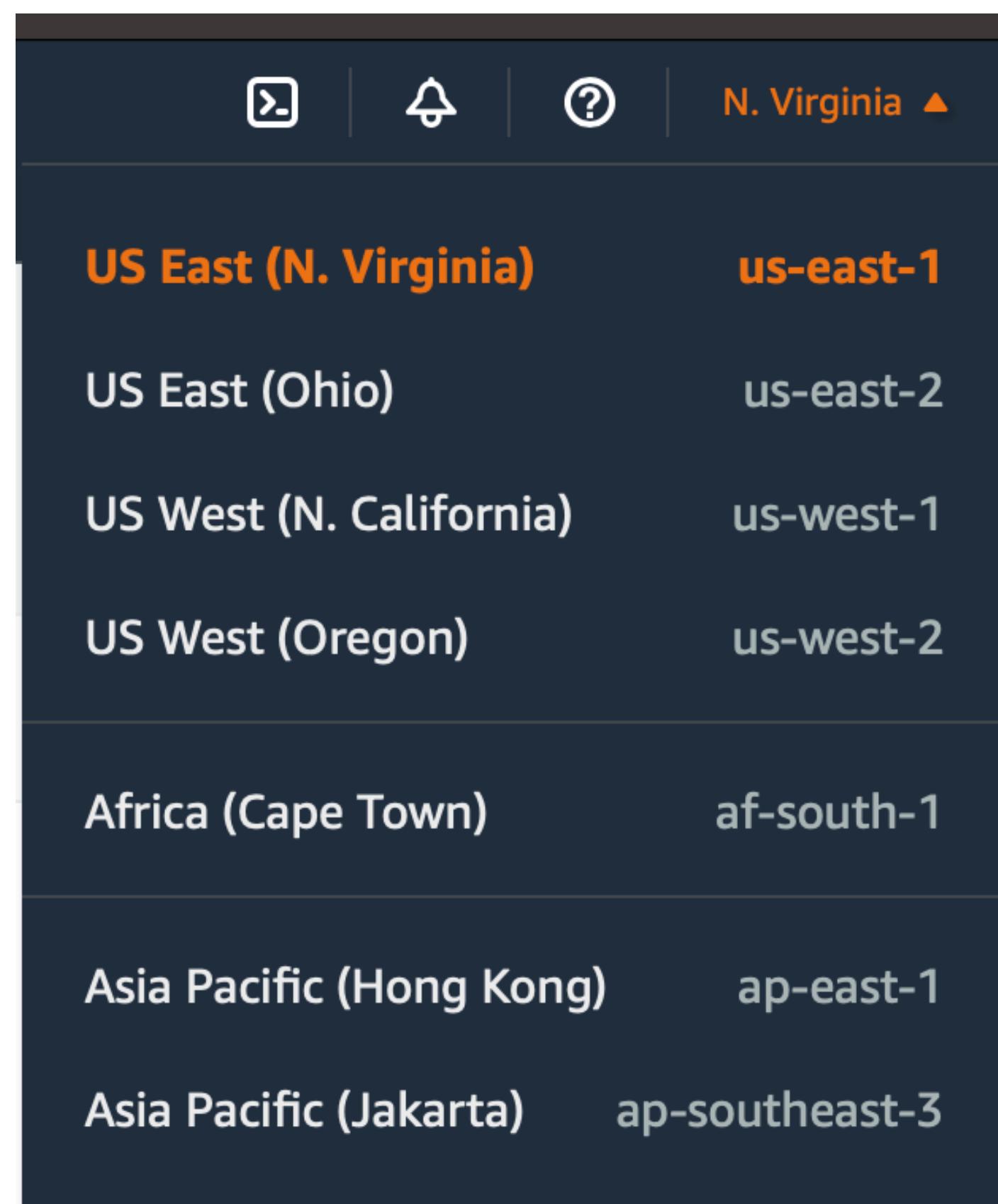
Summary (Top Tab):

DB identifier	CPU	Status	Class
employee	8.17%	Backing-up	db.t2.micro
Role	Current activity	Engine	Region & AZ
Instance	0 Connections	MySQL Community	us-east-1f

Connectivity & security (Bottom Tab):

Endpoint & port	Networking	Security
Endpoint employee.cka3tc4n3juu.us-east-1.rds.amazonaws.com	Availability Zone us-east-1f	VPC security groups default (sg-06ae99d5596dd2e35) <input checked="" type="radio"/> Active
Port 3306	VPC vpc-0dcb52fe71c0a9240	Public accessibility Yes
	Subnet group default-vpc-0dcb52fe71c0a9240	Certificate authority rds-ca-2019
	Subnets subnet-02d48cf10e2eac2d subnet-05c870d7cbf593d55 subnet-0f3147c730bfe38c2 subnet-0711d303af96e9414 subnet-09d9f83ce518bc83a subnet-083c2d1ee788dbb5a	Certificate authority date August 22, 2024, 22:38 (UTC+05:30)

- Note : your s3 bucket origin and RDS origin should be same.
- For example:



STEP 3

CREATE AN S3 SERVICE ON AWS

The screenshot shows the Amazon S3 service console interface. On the left, there's a sidebar with various navigation options like 'Buckets', 'Access Points', and 'Storage Lens'. The main area displays the 'addemployee76' bucket details. The 'Objects' tab is selected, showing a message that says 'Objects (0)' and 'No objects'. It includes buttons for 'Upload' (orange), 'Actions', 'Create folder', 'Delete', 'Download', 'Copy URL', and 'Copy S3 URI'. A search bar at the top says 'Find objects by prefix'. Below the search bar is a table header with columns: Name, Type, Last modified, Size, and Storage class.

Amazon S3

Buckets

Access Points

Object Lambda Access Points

Multi-Region Access Points

Batch Operations

Access analyzer for S3

Block Public Access settings for this account

Storage Lens

Dashboards

AWS Organizations settings

Feature spotlight 3

AWS Marketplace for S3

Amazon S3 > Buckets > addemployee76

addemployee76 Info

Objects Properties Permissions Metrics Management Access Points

Objects (0)

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](#)

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

Find objects by prefix

Name	Type	Last modified	Size	Storage class
No objects				
You don't have any objects in this bucket.				

Upload

STEP 4

CREATE AN EC2 INSTANCE ON AWS (UBUNTU)

The screenshot shows the 'Connect to instance' page for an Ubuntu EC2 instance. The instance ID is i-0dda256ab0203dc46. The 'SSH client' tab is selected. Below it, there's a list of steps to connect:

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is ubuntukeypair.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 ubuntukeypair.pem
4. Connect to your instance using its Public DNS:
ec2-44-211-61-41.compute-1.amazonaws.com

Below the steps is an example command:

```
ssh -i "ubuntukeypair.pem" ubuntu@ec2-44-211-61-41.compute-1.amazonaws.com
```

A note in a box states: "Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name."

- Copy the code from example and run in your terminal

STEP 5

CONECT YOUR EC2 INSTANCE TO YOUR TERMINAL

```
● ● ● Downloads — ubuntu@ip-172-31-92-57: ~ — ssh -i ubuntukeypair.pem ubuntu@ec2-44-211-61-41.compute-1.amazonaws.com — 109x33

yesh@Gs-MacBook-Air Downloads % ssh -i "ubuntukeypair.pem" ubuntu@ec2-44-211-61-41.compute-1.amazonaws.com
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1019-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 System information as of Sun Nov  6 06:31:02 UTC 2022

 System load:  0.0          Processes:      99
 Usage of /:   19.6% of 7.57GB  Users logged in:  0
 Memory usage: 21%          IPv4 address for eth0: 172.31.92.57
 Swap usage:   0%

0 updates can be applied immediately.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-92-57:~$
```

STEP 6

INSTALL MYSQL ON AWS EC2(UBUNTU)

- command “sudo apt-get install mysql-client”

```
ubuntu@ip-172-31-92-57:~$  
ubuntu@ip-172-31-92-57:~$ sudo apt-get install mysql-client  
[Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  
  mysql-client-8.0 mysql-client-core-8.0 mysql-common  
The following NEW packages will be installed:  
  mysql-client mysql-client-8.0 mysql-client-core-8.0 mysql-common  
0 upgraded, 4 newly installed, 0 to remove and 78 not upgraded.  
Need to get 2702 kB of archives.  
After this operation, 62.3 MB of additional disk space will be used.  
Do you want to continue? [Y/n] y  
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-client-core-8.0 amd64 8.0.31-0ubuntu0.22.04.1 [2663 kB]  
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 mysql-common all 5.8+1.0.8 [7212 B]  
|Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-client-8.0 amd64 8.0.31-0ubuntu0.22.04.1 [22.7 kB]  
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-client all 8.0.31-0ubuntu0.22.04.1 [9428 B]  
Fetched 2702 kB in 0s (25.6 MB/s)  
Selecting previously unselected package mysql-client-core-8.0.  
(Reading database ... 63663 files and directories currently installed.)  
Preparing to unpack .../mysql-client-core-8.0_8.0.31-0ubuntu0.22.04.1_amd64.deb ...  
Unpacking mysql-client-core-8.0 (8.0.31-0ubuntu0.22.04.1) ...  
Selecting previously unselected package mysql-common.  
Preparing to unpack .../mysql-common_5.8+1.0.8_all.deb ...  
Unpacking mysql-common (5.8+1.0.8) ...  
Selecting previously unselected package mysql-client-8.0.  
Preparing to unpack .../mysql-client-8.0_8.0.31-0ubuntu0.22.04.1_amd64.deb ...  
Unpacking mysql-client-8.0 (8.0.31-0ubuntu0.22.04.1) ...  
Selecting previously unselected package mysql-client.  
Preparing to unpack .../mysql-client_8.0.31-0ubuntu0.22.04.1_all.deb ...  
Unpacking mysql-client (8.0.31-0ubuntu0.22.04.1) ...  
Setting up mysql-common (5.8+1.0.8) ...  
update-alternatives: using /etc/mysql/my.cnf.fallback to provide /etc/mysql/my.cnf (my.cnf) in auto mode  
Setting up mysql-client-core-8.0 (8.0.31-0ubuntu0.22.04.1) ...  
Setting up mysql-client-8.0 (8.0.31-0ubuntu0.22.04.1) ...  
Setting up mysql-client (8.0.31-0ubuntu0.22.04.1) ...  
Processing triggers for man-db (2.10.2-1) ...  
Scanning processes...  
Scanning linux images...  
  
Running kernel seems to be up-to-date.  
  
No services need to be restarted.  
  
No containers need to be restarted.  
  
No user sessions are running outdated binaries.  
  
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
ubuntu@ip-172-31-92-57:~$
```

STEP 7

CONNECT RDS TO EC2(ubuntu)

- Command “mysql -h <endpoint> -u <master username> -p” press enter and then enter your master password and proceed.

```
ubuntu@ip-172-31-92-57:~$  
ubuntu@ip-172-31-92-57:~$ mysql -h employee.cka3tc4n3juu.us-east-1.rds.amazonaws.com -u admin -p  
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 15  
Server version: 8.0.28 Source distribution  
  
Copyright (c) 2000, 2022, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
mysql> █
```

STEP 8

CREATE A DATABASE EMPLOYEE IN MYSQL

- QUERY “create database employee;”

```
[mysql]>
[mysql]> create database employee;
Query OK, 1 row affected (0.01 sec)

mysql> █
```

STEP 9

CREATE A TABLE EMPLOYEE

```
[mysql]>
[mysql]> create table employee(
[    -> emp_id varchar(20),
[    -> first_name varchar(20),
[    -> last_name varchar(20),
[    -> pri_skil varchar(20),
[    -> location varchar(20));
Query OK, 0 rows affected (0.04 sec)

[mysql]>
[mysql]>
[mysql]> desc 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    |          |
| pri_skil   | varchar(20) | YES  |     | NULL    |          |
| location   | varchar(20) | YES  |     | NULL    |          |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> █
```

Change configuration

Change in config.py file

- customhost = "endpoint"
- customuser = "master username "
- custompass = "password"
- customdb = "db instance identifier name"
- custombucket = "s3 bucket name"
- customregion = "region"
-

STEP 10

Configure like this config.py file

```
● ● ● *config.py - /Users/yesh/Desktop/AWS/aws-live-master/config.py (3.10.1)*  
customhost = "employee.cka3tc4n3juu.us-east-1.rds.amazonaws.com"  
customuser = "admin"  
custompass = "123456789"  
customdb = "employee"  
custombucket = "addemployee76"  
customregion = "us-east-1b"
```

Install following in your terminal

- sudo apt-get update
- # For Sql-client
- sudo apt-get install mysql-client
- # For python and related frameworks
- sudo apt-get install python3
- Sudo apt-get install python3-pip
- sudo apt-get install python3-flask
- sudo apt-get install python3-pymysql
- sudo apt-get install python3-boto3
- # for running application
- sudo python3 Empapp.py

STEP 11

CLONE THE FILE FROM GITHUB

- COMMAND “git clone https://github.com/hshar94/aws-live.git”

```
ubuntu@ip-172-31-92-57:~$ git clone https://github.com/hshar94/aws-live.git
Cloning into 'aws-live'...
remote: Enumerating objects: 42, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 42 (delta 3), reused 3 (delta 3), pack-reused 29
Receiving objects: 100% (42/42), 11.36 KiB | 2.84 MiB/s, done.
Resolving deltas: 100% (13/13), done.
ubuntu@ip-172-31-92-57:~$
```

STEP 12

RUN PYTHON FILE “EmpApp.py”

- Command “sudo python3 EmpApp.py”.

```
[ubuntu@ip-172-31-92-57:~$ ls
aws-live
[ubuntu@ip-172-31-92-57:~$ cd aws-live
[ubuntu@ip-172-31-92-57:~/aws-live$ ls
EmpApp.py  __pycache__  config.py  readme  templates
[ubuntu@ip-172-31-92-57:~/aws-live$ vim config.py
[ubuntu@ip-172-31-92-57:~/aws-live$
[ubuntu@ip-172-31-92-57:~/aws-live$ sudo python3 EmpApp.py
 * Serving Flask app 'EmpApp' (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on all addresses.
   WARNING: This is a development server. Do not use it in a production deployment.
 * Running on http://172.31.92.57:80/ (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 216-537-139
```

STEP 13

Copy the link and paste in your browser

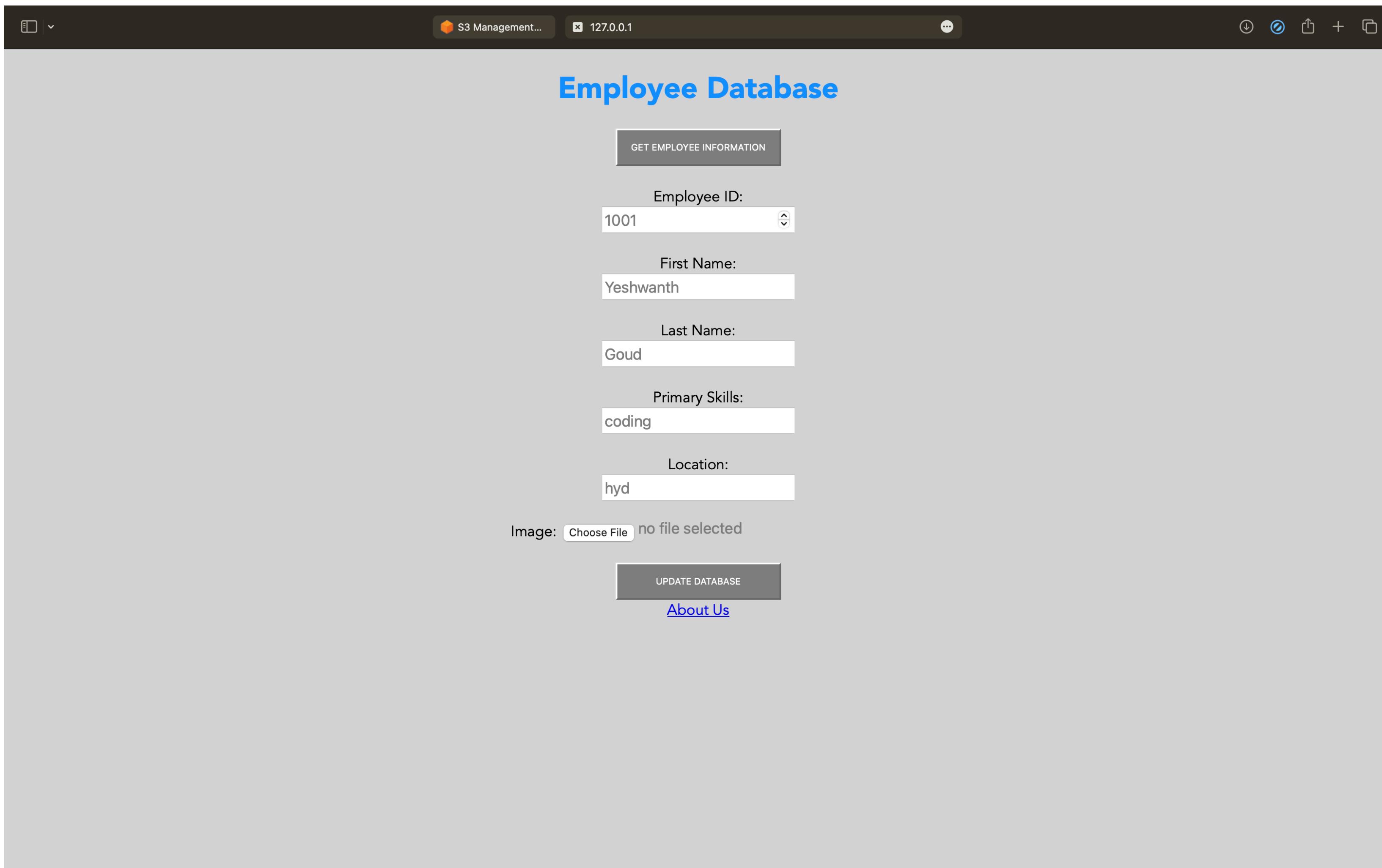
- Link “<http://127.0.0.1:80>”

```
[yesh@Gs-MacBook-Air aws-live-master % sudo python3 EmpApp.py
[Password:
 * Serving Flask app 'EmpApp'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:80
 * Running on http://192.168.0.101:80
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 487-689-250
```

DEPLOYED SUCESSFULLY

STEP 14

ADD DETAILS OF YOUR EMPLOYEE ON WEBSITE



The screenshot shows a web browser window with the title "Employee Database". The browser's address bar displays "127.0.0.1". The main content area contains the following fields:

- Employee ID:** A dropdown menu set to "1001".
- First Name:** An input field containing "Yeshwanth".
- Last Name:** An input field containing "Goud".
- Primary Skills:** An input field containing "coding".
- Location:** An input field containing "hyd".
- Image:** A file input field labeled "Choose File" with the status "no file selected".
- Buttons:** A "GET EMPLOYEE INFORMATION" button at the top, an "UPDATE DATABASE" button at the bottom, and a link "About Us" below the update button.

NOW DATA IS SUCESSFULLY INSERTED INTO MYSQL DATABASE

```
aws-live-master — Python ▾ sudo — 103x35
[Password:
 * Serving Flask app 'EmpApp'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI
server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:80
 * Running on http://192.168.0.101:80
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 487-689-250
127.0.0.1 - - [06/Nov/2022 18:08:45] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [06/Nov/2022 18:08:45] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:09:43] "POST /adddemp HTTP/1.1" 200 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /adddemp HTTP/1.1" 405 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon-precomposed.png HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon.png HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon-precomposed.png HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon.png HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /adddemp HTTP/1.1" 405 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon-precomposed.png HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon.png HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon-precomposed.png HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon.png HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon-precomposed.png HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [06/Nov/2022 18:16:39] "GET /apple-touch-icon.png HTTP/1.1" 404 -
Data inserted in MySQL RDS... uploading image to S3...
127.0.0.1 - - [06/Nov/2022 18:17:22] "POST /adddemp HTTP/1.1" 200 -
```

DATA IS ALSO SUCESSFULLY INSERTED INTO TABLE

```
[mysql]> select * from employee;
+-----+-----+-----+-----+
| emp_id | first_name | last_name | pri_skill | location |
+-----+-----+-----+-----+
| 1001   | Yeshwanth | Goud      | coding    | hyd       |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql>
```

LETS INSERT SOME OTHER DATA

Employee Database

Employee ID:
1003

First Name:
Aakash

Last Name:
Vardhan

Primary Skills:
coding

Location:
Sangareddy

Image: Screenshot 202....09.46 PM.png

[About Us](#)

Employee Database

Employee ID:
1006

First Name:
Joseph

Last Name:
massive

Primary Skills:
coding

Location:
tanzania

Image: Screenshot 202....09.46 PM.p

[About Us](#)

Employee Database

Employee ID:
1005

First Name:
Govardan

Last Name:
darsi

Primary Skills:
coding

Location:
Chandannagar

Image: Screenshot 202....09.46 PM.png

[About Us](#)

Employee Database

Employee ID:
1004

First Name:
Praneeth

Last Name:
kalivemula

Primary Skills:
coding

Location:
Shankarpalli

Image: Screenshot 202....09.46 PM.png

[About Us](#)

Employee Database

Employee ID:
1002

First Name:
Jagadish

Last Name:
chilakala

Primary Skills:
coding

Location:
Kondapur

Image: Screenshot 202....09.46 PM.png

[About Us](#)

DATA IS INSERTED INTO MYSQL RDS

```
127.0.0.1 -- [06/Nov/2022 18:16:39] "GET /apple-touch-icon.png HTTP/1.1" 404 -  
127.0.0.1 -- [06/Nov/2022 18:16:39] "GET /apple-touch-icon-precomposed.png HTTP/1.1" 404 -  
127.0.0.1 -- [06/Nov/2022 18:16:39] "GET /favicon.ico HTTP/1.1" 404 -  
127.0.0.1 -- [06/Nov/2022 18:16:39] "GET /apple-touch-icon.png HTTP/1.1" 404 -  
Data inserted in MySQL RDS... uploading image to S3...  
127.0.0.1 -- [06/Nov/2022 18:17:22] "POST /adtemp HTTP/1.1" 200 -  
127.0.0.1 -- [06/Nov/2022 18:45:11] "GET / HTTP/1.1" 200 -  
Data inserted in MySQL RDS... uploading image to S3...  
127.0.0.1 -- [06/Nov/2022 18:45:49] "POST /adtemp HTTP/1.1" 200 -  
127.0.0.1 -- [06/Nov/2022 18:46:52] "GET / HTTP/1.1" 200 -  
Data inserted in MySQL RDS... uploading image to S3...  
127.0.0.1 -- [06/Nov/2022 18:47:25] "POST /adtemp HTTP/1.1" 200 -  
Data inserted in MySQL RDS... uploading image to S3...  
127.0.0.1 -- [06/Nov/2022 18:50:14] "POST /adtemp HTTP/1.1" 200 -  
Data inserted in MySQL RDS... uploading image to S3...  
127.0.0.1 -- [06/Nov/2022 18:51:03] "POST /adtemp HTTP/1.1" 200 -  
Data inserted in MySQL RDS... uploading image to S3...  
127.0.0.1 -- [06/Nov/2022 18:51:52] "POST /adtemp HTTP/1.1" 200 -  
Data inserted in MySQL RDS... uploading image to S3...  
127.0.0.1 -- [06/Nov/2022 18:54:17] "POST /adtemp HTTP/1.1" 200 -  
^C%
```

ALL THE DATA IS AUTOMATICALLY IMPORTED TO EMPLOYEE TABLE

```
mysql>
mysql> select * from employee;
+-----+-----+-----+-----+-----+
| emp_id | first_name | last_name | pri_skill | location |
+-----+-----+-----+-----+-----+
| 1001   | Yeshwanth  | Goud      | coding    | hyd       |
| 1002   | Jagadish   | chilakala | coding    | Kondapur  |
| 1003   | Aakash     | Vardhan   | coding    | Sangareddy |
| 1004   | Praneeth   | kalivemula | coding   | Shankarpalli |
| 1005   | Govardan   | darsi     | coding    | Chandannagar |
| 1006   | Joseph     | massive   | coding    | tanzania  |
+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```

SUCESSFULLY DEPLOYED AWS LIVE PROJECT

**DONE BY
-G.YESHWANTH GOUD(HU21CSC10200001)
BSC.CSCS**