Serverless computing using AWS LAMBDA

Overview:

1.Deploy code in AWS lambda

2.Create and invoke API for lambda function

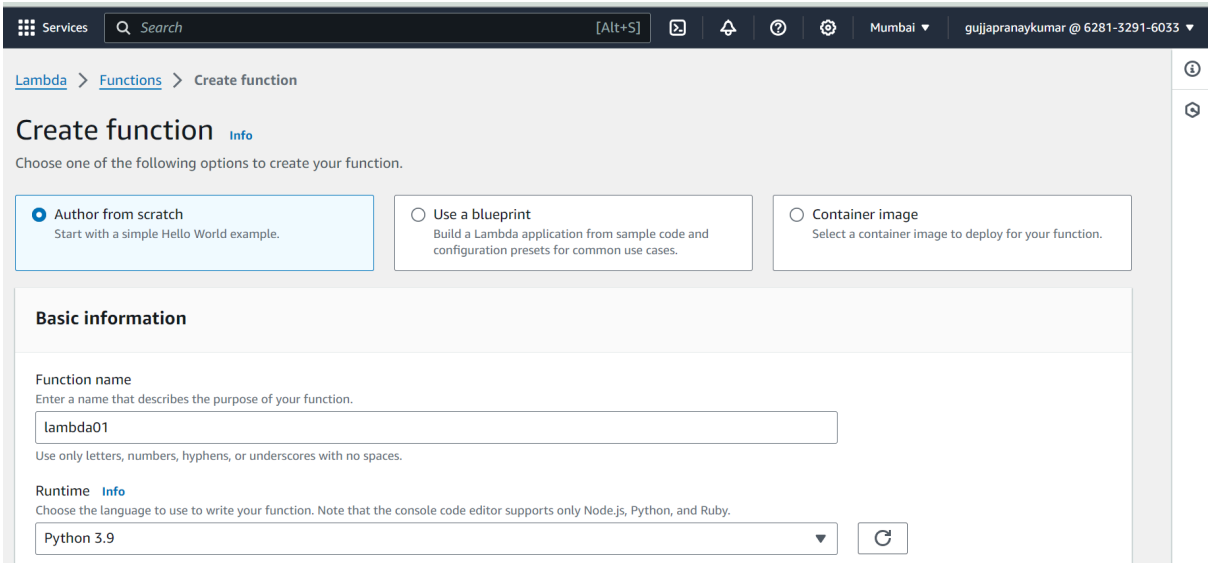
Code deployment using AWS Lambda

1.Navigate to Lambda page and click on create function

2.Provide the function name

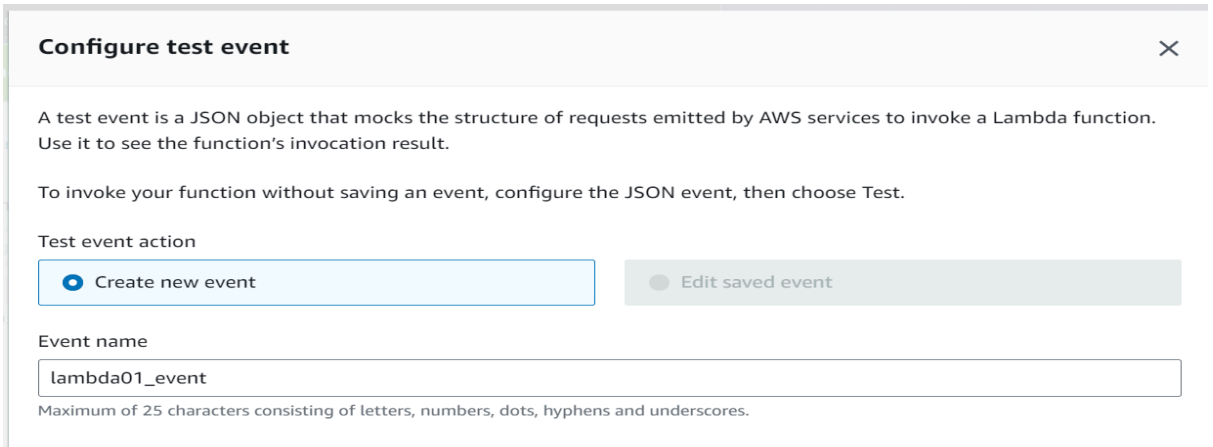
3.Select the Runtime as python 3.9

4.Click on create function and it should be created successfully
A:



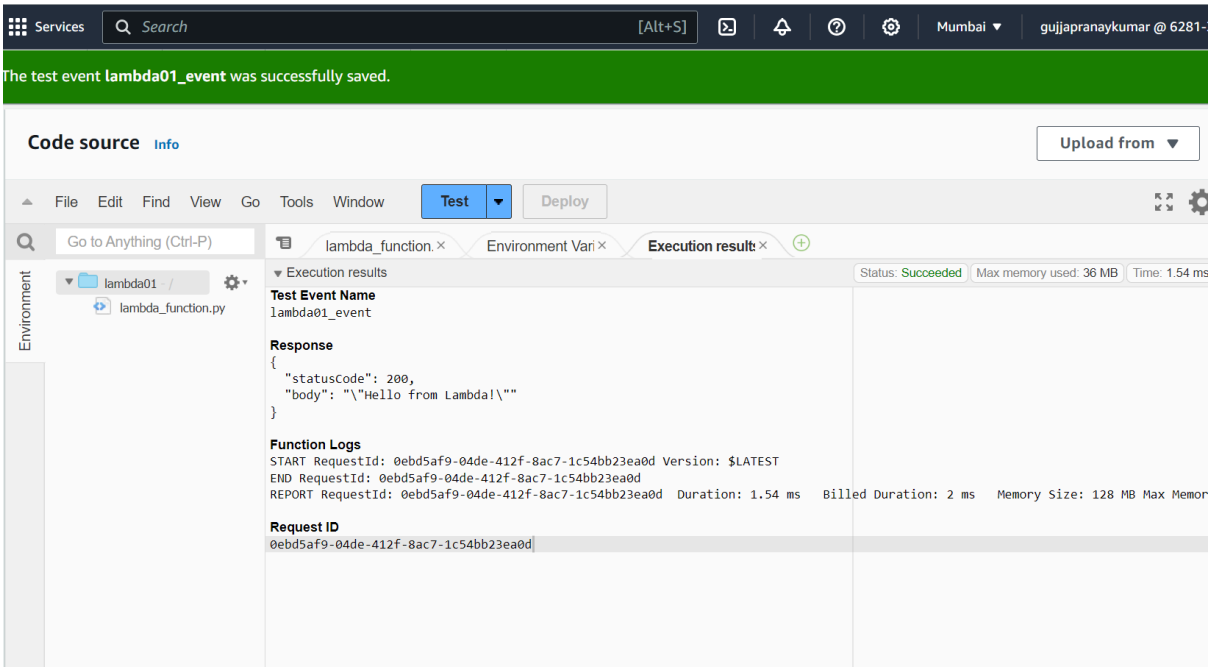
5.Scroll down and you can see a sample python code that you can use it for the deployment

6.Go to test tab to create a new event, provide the event name and click on save event and the event will get saved
A:



7.Now navigate back to the code tab and click on test

8.You will get the execution results as below
A:

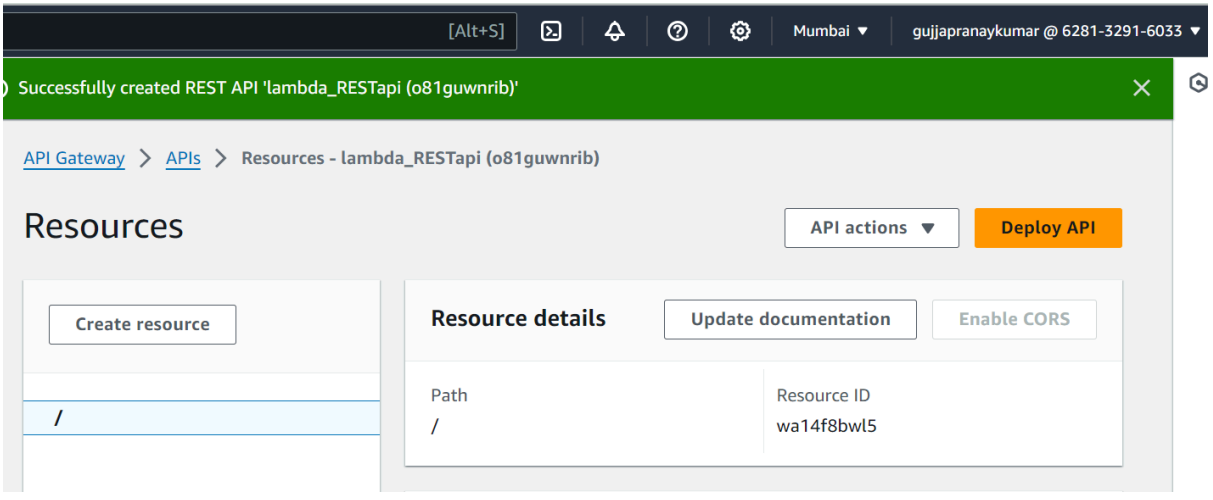


Configure REST API to call the lambda function

1.Navigate to API gateway dashboard

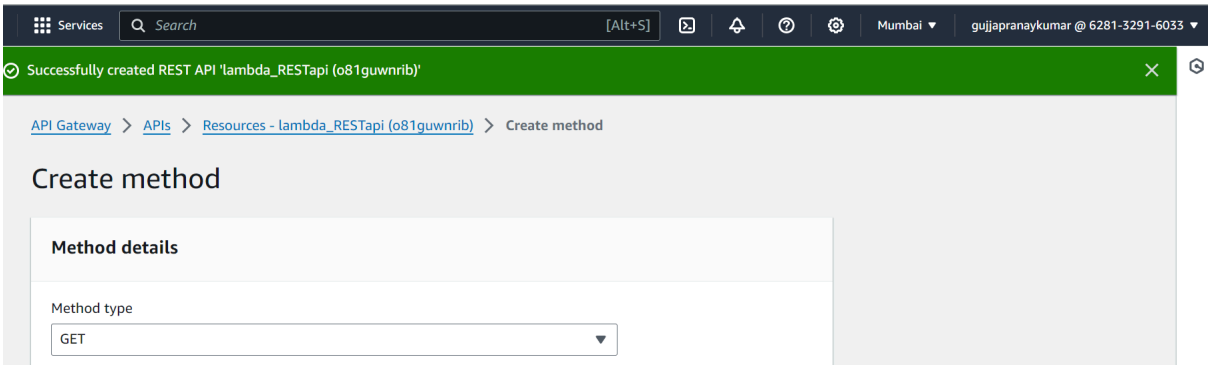
2.Scroll down and select Build option on REST api in the API type

3.Click on New api, provide the api name, description and click on create API you will get a response as below
A:



4.Navigate to Actions and click on ‘Create Method’ and select GET from the resource dropdown and click on tick mark

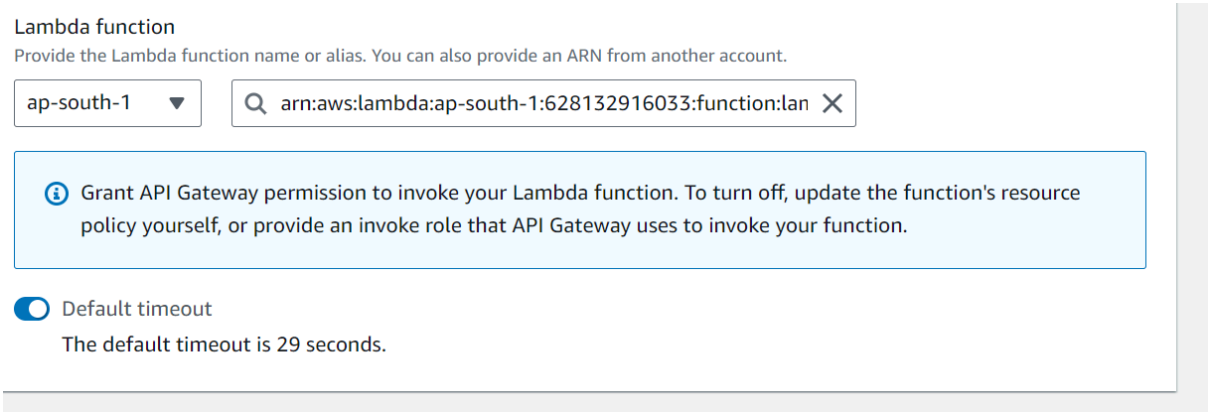
A:



5.In the LAMBDA function text box type the lambda function that you have created before and leave remaining all settings as default

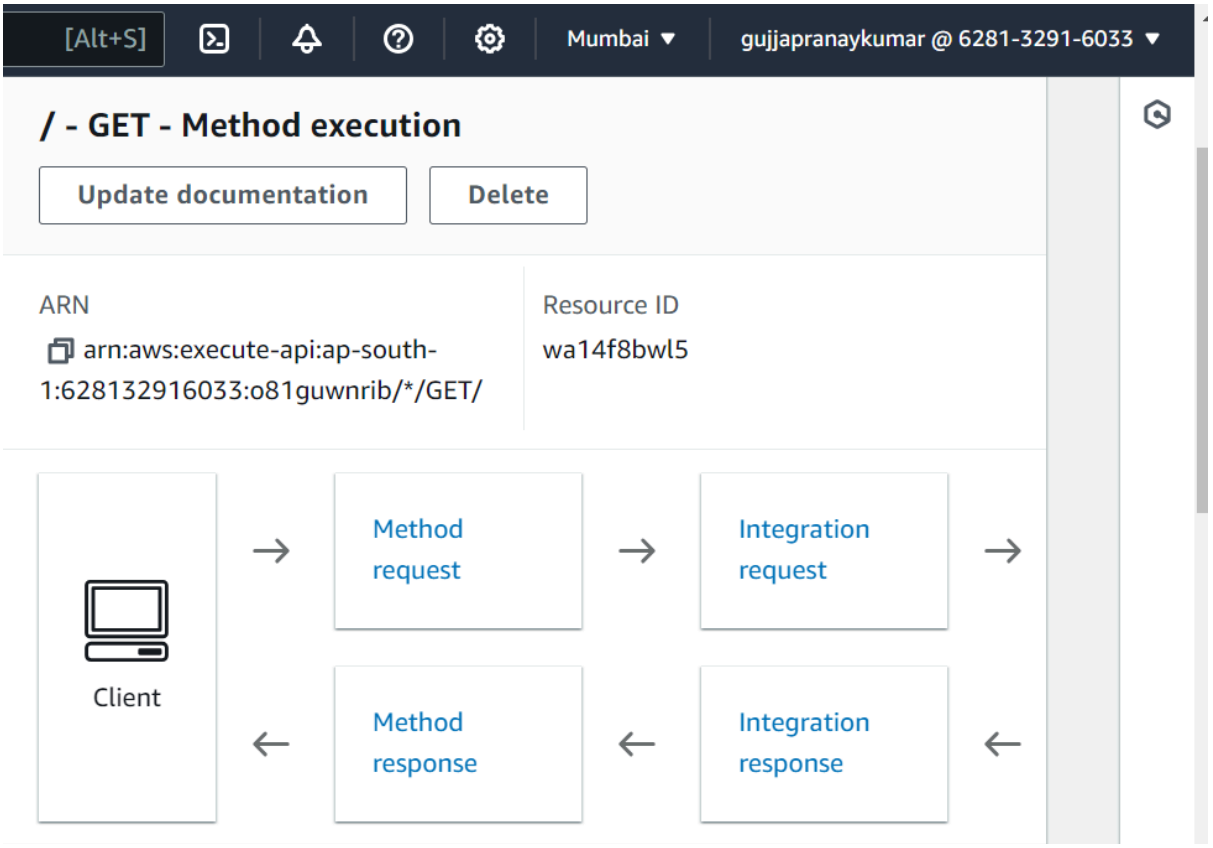
6.Click on Save and then click ‘OK’ in the prompt

A:



7.Resource will be created and you will get a response as below

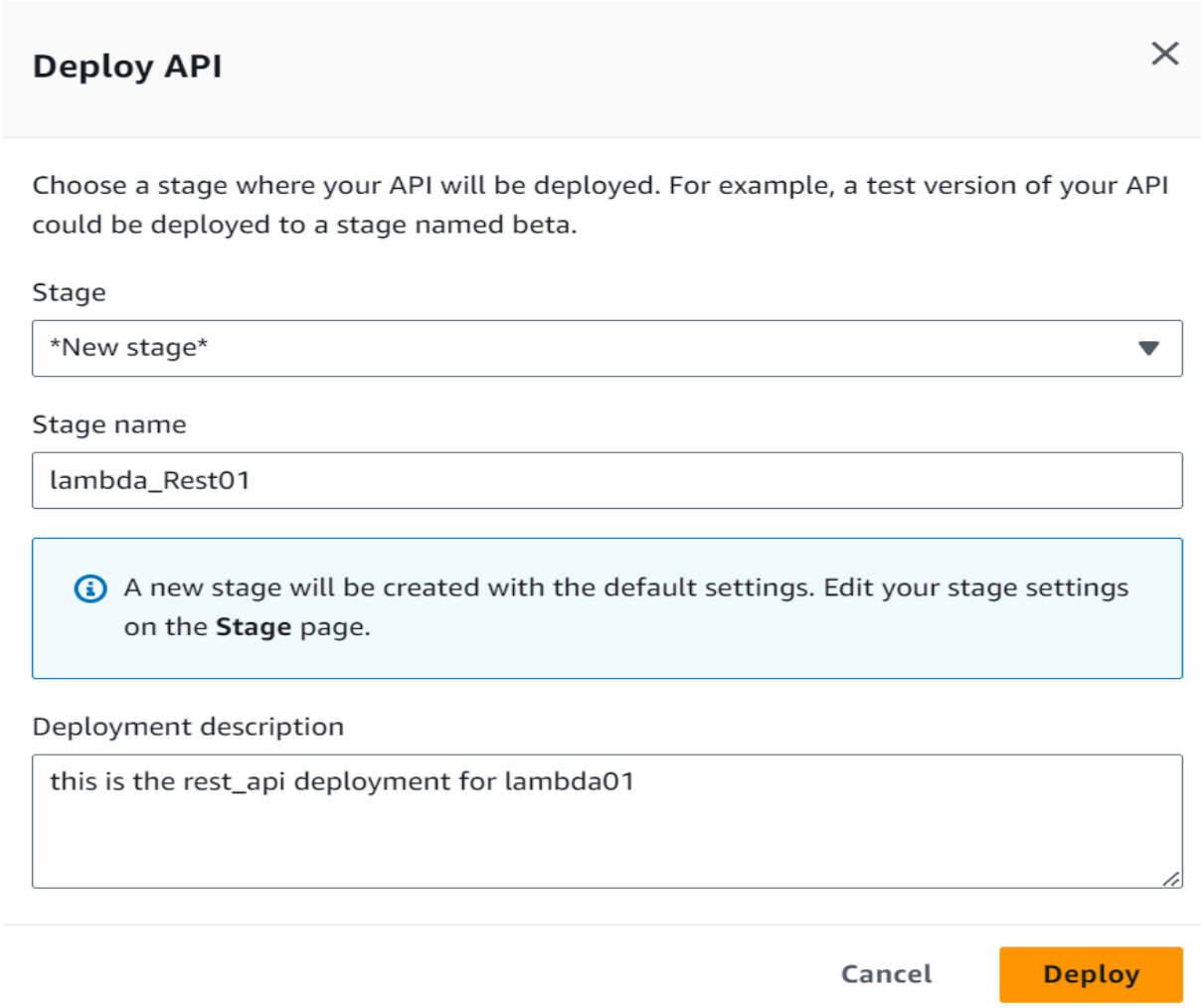
A:



8.Click on Actions and click on Deploy API

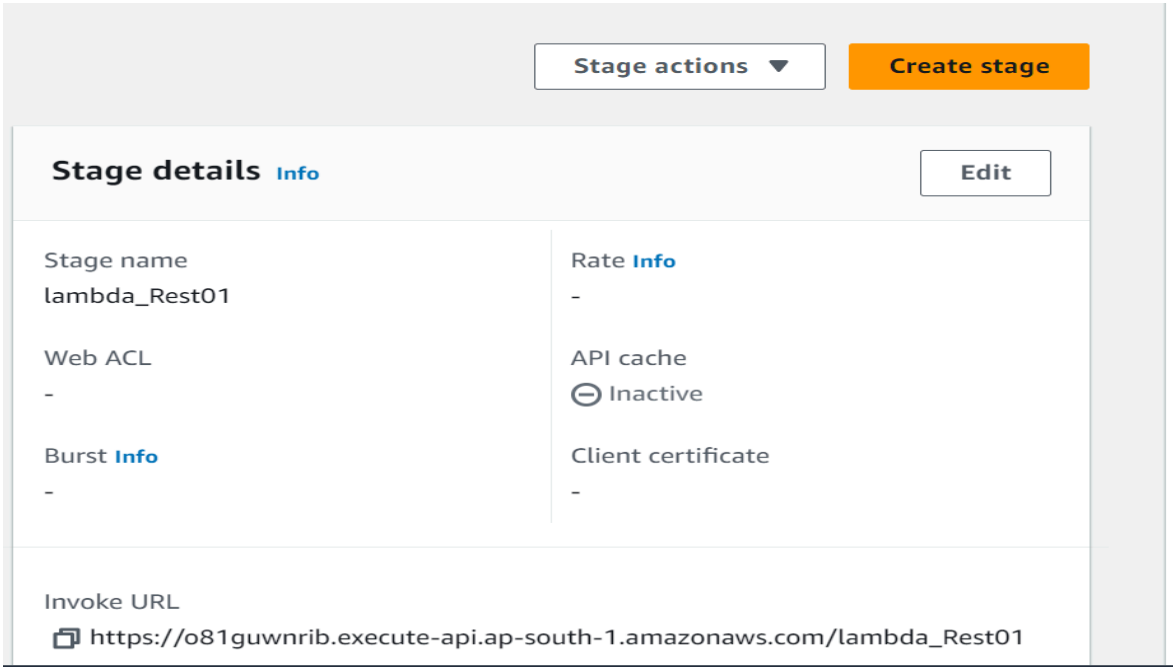
9.Select the Deployment stage as New stage, provide the stage name, stage description, and deployment description and click on deploy

A:



10. You will get a response page as below

A:



11. Now click on the Invoke url and you will get the function invoked using REST API
A:

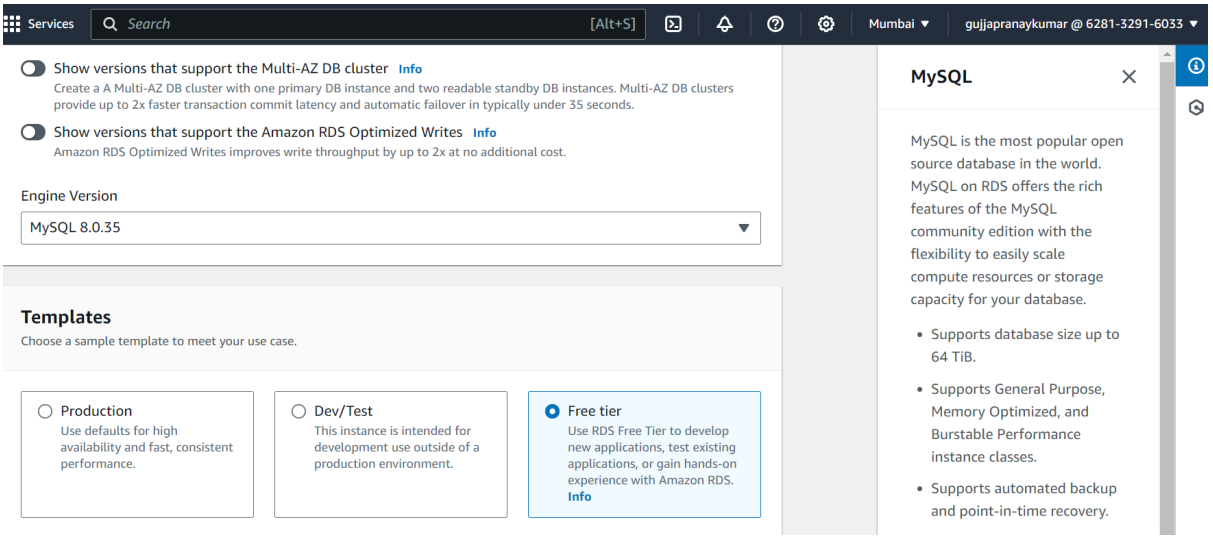
END

Creating Lambda Function to access RDS Database

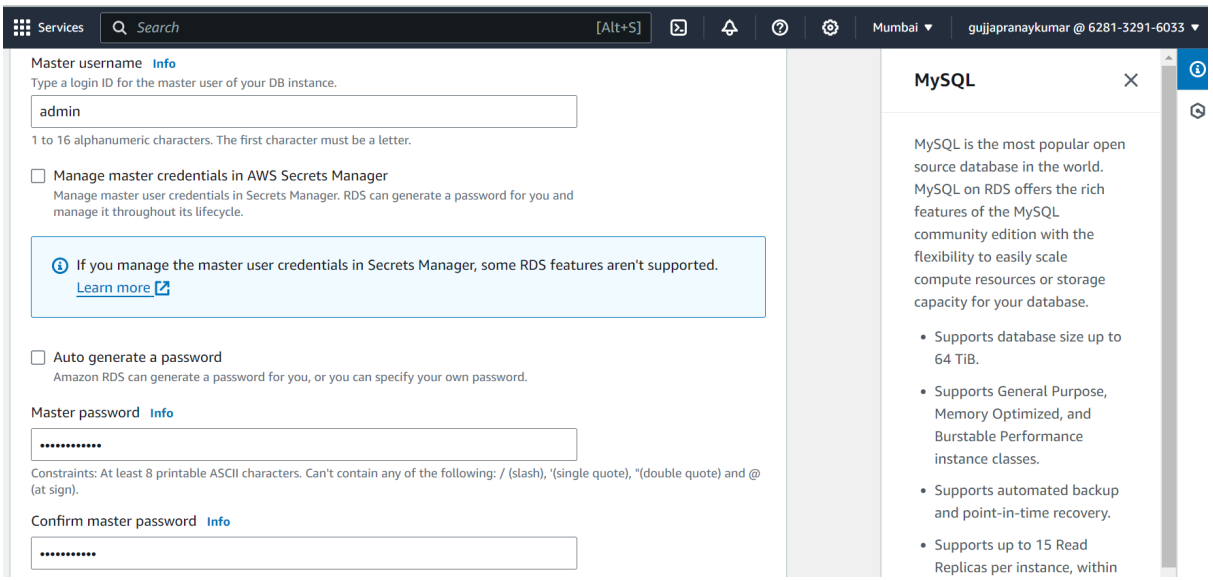
Creating RDS Database

1.Navigate to RDS service and Create a Database

2.Choose Standard create, Engine type MySQL, Leave Engine version as default, choose Templates as Free tier
A:

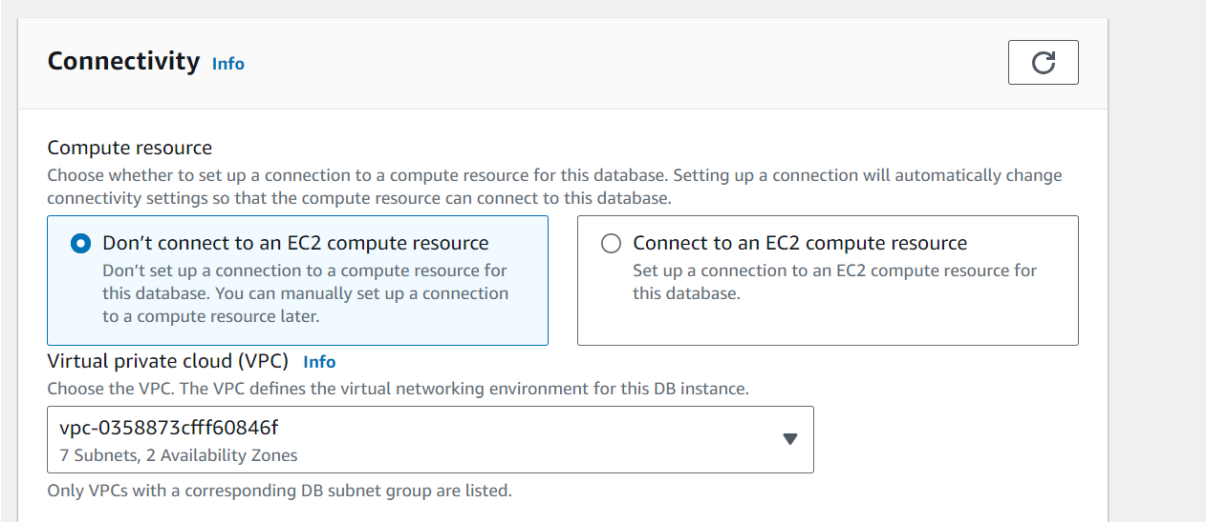


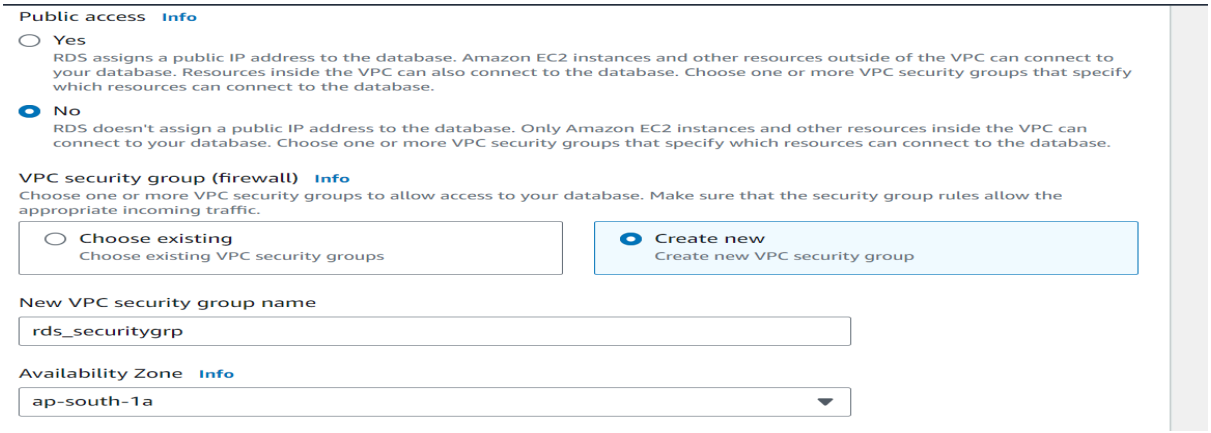
3.In Settings Give a name for your Database or leave it as default, leave Master username as default,Set Master password for your Database
A:



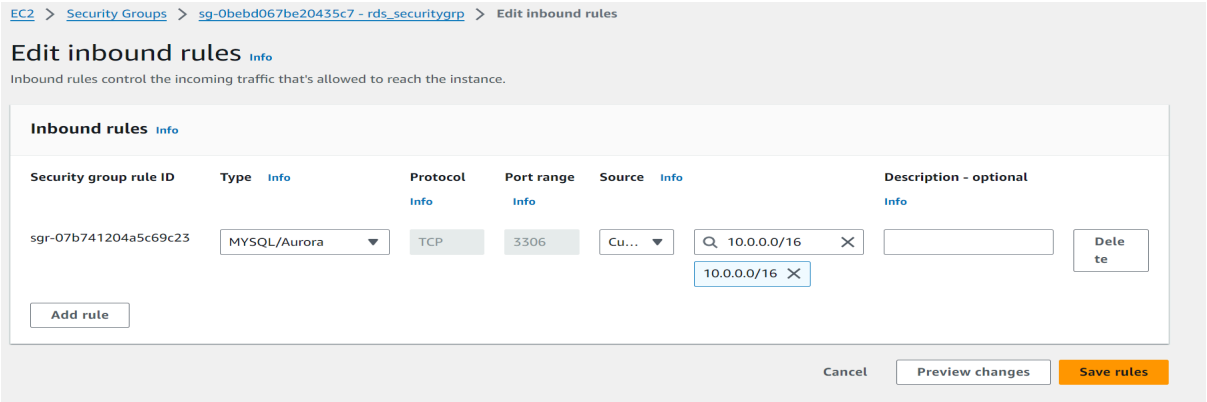
4.Instance Configuration and Storage you can leave it as default

5.In Connectivity Choose first option(Don’t connect to an EC2 Compute Resource) and in VPC select your VPC, Public access No, VPC security groups, you can create a new security group
A:





6.In Database Authentication choose first method
7.Disable Monitoring and click on Create database, Your database will be created
8.Go to the database and open your security group, In inbound rules, add a rule in security group, choose the type as MYSQL/Aurora and allow your VPC range, In Outbound rule allow all traffic
A:



Connect to RDS instance

1.Launch an ec2 instance and install mysql alone there follow mattermost software Installation Document

2.Connect to mysql using the command as follows,
mysql -h your-rds-dnsname -u admin -p

3.For Creating Database, tables, Inserting Content into the tables, follow the attached images
A:

```
mysql> create database sampleDB;
Query OK, 1 row affected (0.00 sec)

mysql> use sampleDB;
Database changed
mysql> create table customers(id int not null,name varchar(20) not null,age int not null,address char(25),salary decimal
(18,2),primary key (id));
Query OK, 0 rows affected (0.02 sec)

mysql> show tables;
+--------------------+
| Tables_in_sampleDB |
+--------------------+
| customers          |
+--------------------+
1 row in set (0.00 sec)

mysql> desc customers;
+---------+---------------+------+-----+---------+-------+
| Field   | Type          | Null | Key | Default | Extra |
+---------+---------------+------+-----+---------+-------+
| id      | int(11)       | NO   | PRI | NULL    |       |
| name    | varchar(20)   | NO   |     | NULL    |       |
| age     | int(11)       | NO   |     | NULL    |       |
| address | char(25)      | YES  |     | NULL    |       |
| salary  | decimal(18,2) | YES  |     | NULL    |       |
+---------+---------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

```
mysql> select * from customers;
+----+----------+-----+-----------+---------+
| id | name     | age | address   | salary  |
+----+----------+-----+-----------+---------+
|  1 | Ramesh   |  32 | Ahmedabad | 2000.00 |
|  2 | Khilan   |  25 | Delhi     | 1500.00 |
|  3 | kaushik  |  23 | Kota      | 2000.00 |
|  4 | Chaitali |  25 | Mumbai    | 6500.00 |
|  5 | Hardik   |  27 | Bhopal    | 8500.00 |
+----+----------+-----+-----------+---------+
5 rows in set (0.00 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)
```

Gave privileges to sampleDB to user admin: grant all privileges sampleDB.* to 'admin'@'%';

Creating a Lambda Function

1.Navigate to Lambda Function and create a function and copy the function in the attached document and make all the necessary changes
Lambda Function for RDS DB.pdf

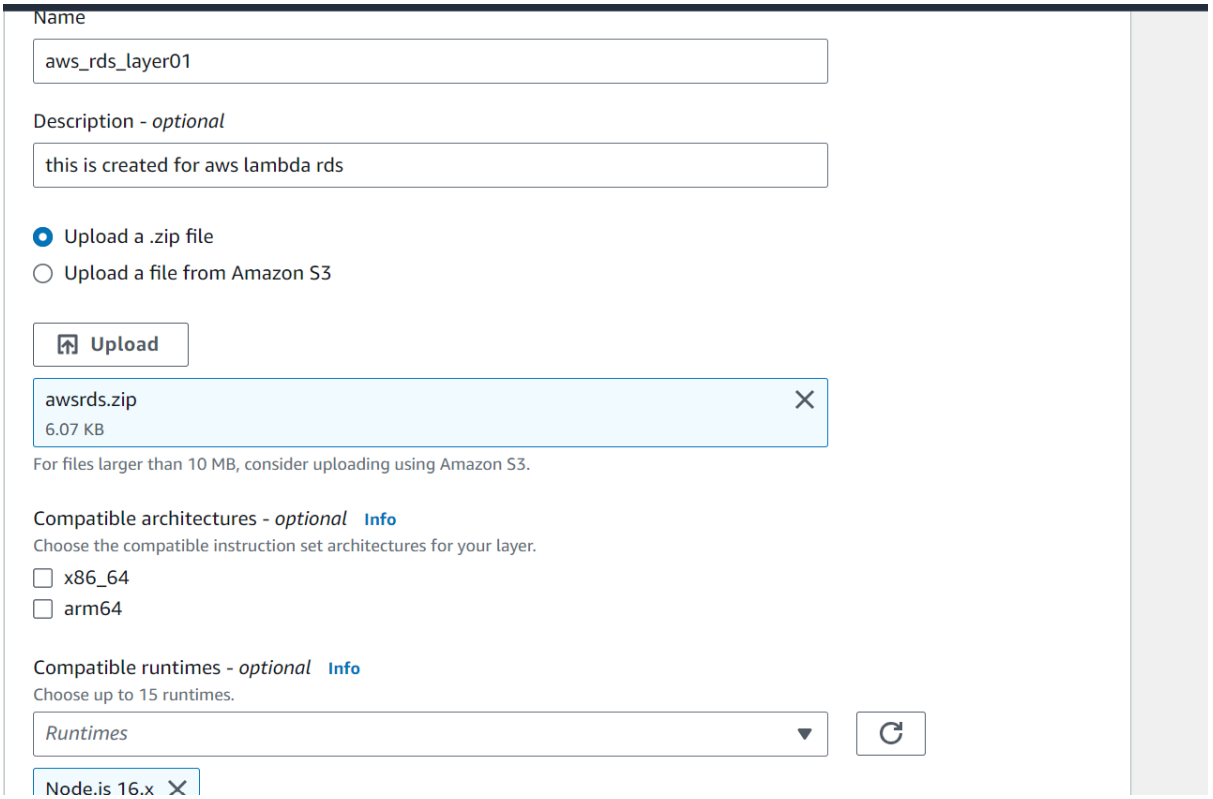
2.You need to have aws-sdk libraries, if you want to execute lambda functions, we can install the aws-sdk using npm, For that go to your centos machine in which you have nodejs and npm, create one folder in your CentOS and navigate into the folder and execute npm install aws-sdk and npm install mysql , You will get node_modules and few files, Create a Zip file for that folders and files. Using filezilla, send that zip file to your windows.
A: the zip file name is nodejs

```
[pranay@localhost aws01]$ sudo zip awsrds.zip node_modules package.json package-lock.json
[sudo] password for pranay:
  adding: node_modules/ (stored 0%)
  adding: package.json (deflated 16%)
  adding: package-lock.json (deflated 83%)
You have new mail in /var/spool/mail/pranay
[pranay@localhost aws01]$ ls
awsrds.zip  node_modules  package.json  package-lock.json
```

3.Add layers to your lambda functions. Navigate to layers and click on create layer

4.Give a layer name and upload that zip file which you have created before

5.Choose the run time as Nodejs16
A:



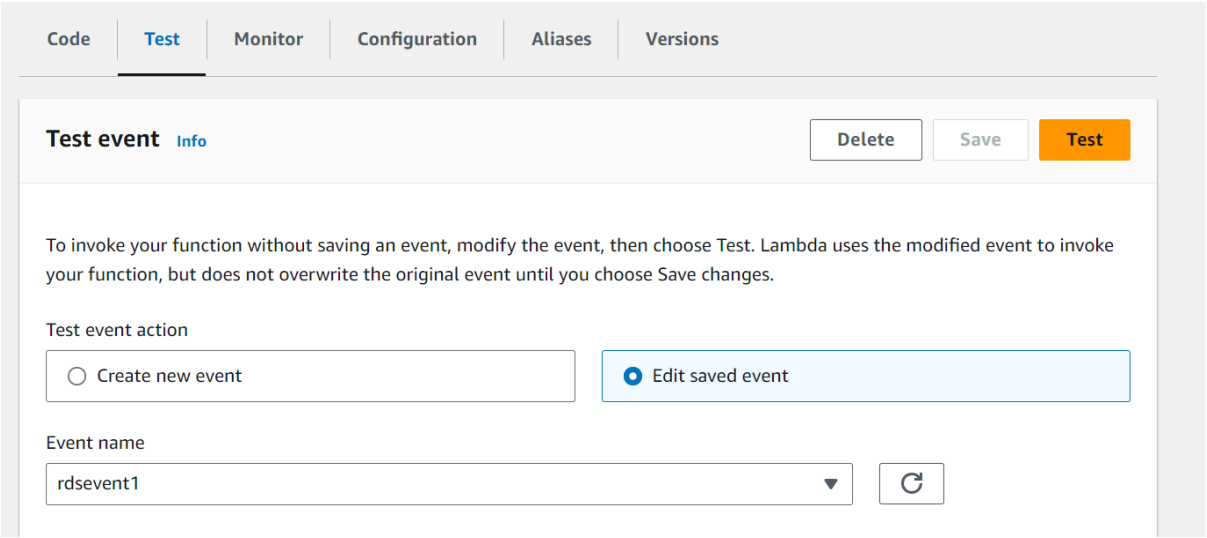
6.In the lambda Configuration choose your VPC, Subnet, Security groups Which will make your lambda available at your VPC level,

A:

7.In the lambda Configuration, In Permissions, make sure lambda has correct roles to access your Database, If not create a role and attach it to the lambda
A: gave full access to ec2 and vpc

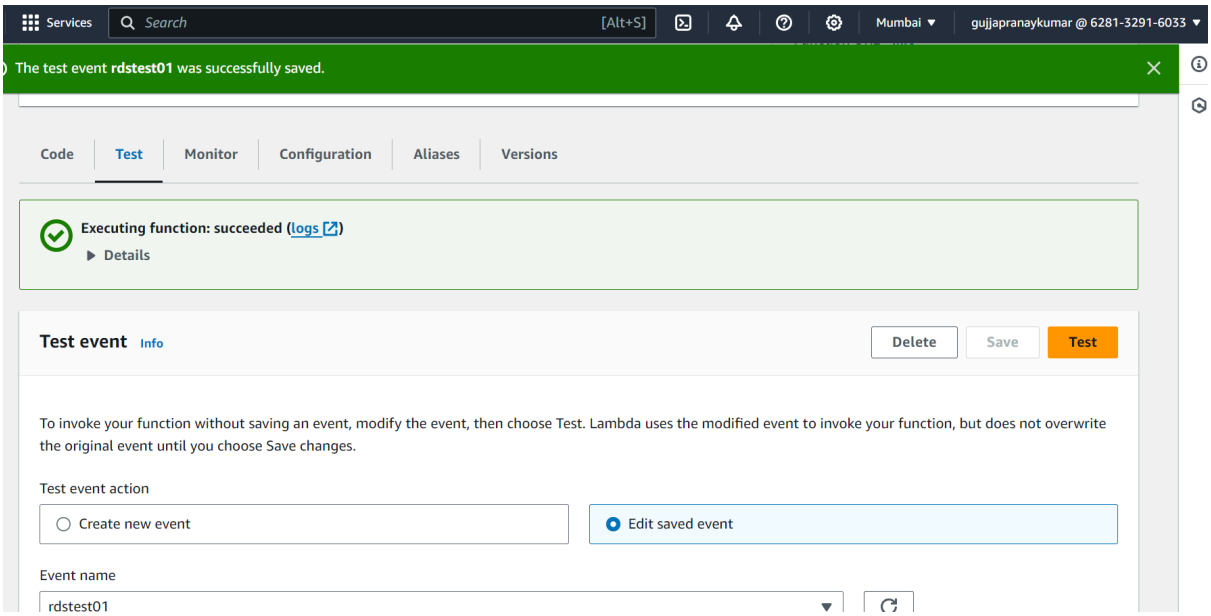
Testing Lambda Function

1.Test your lambda function by creating a event and save it and click on test.
A:



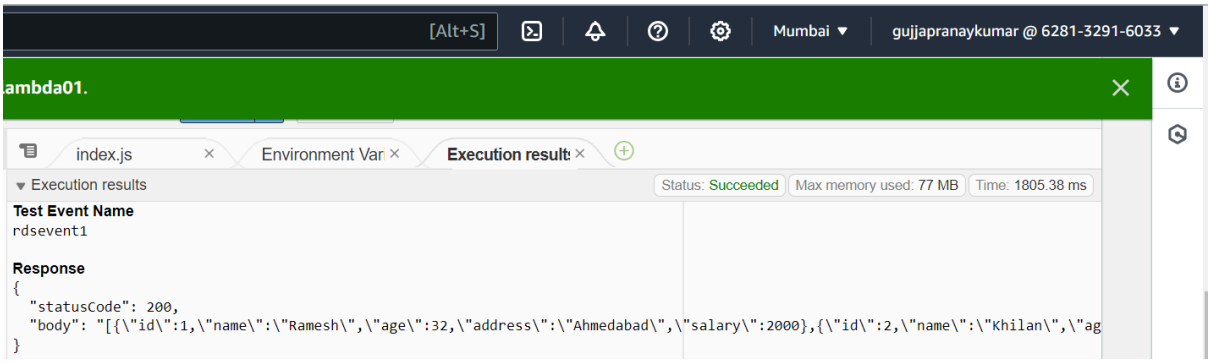
2.Make sure you got no error

A:



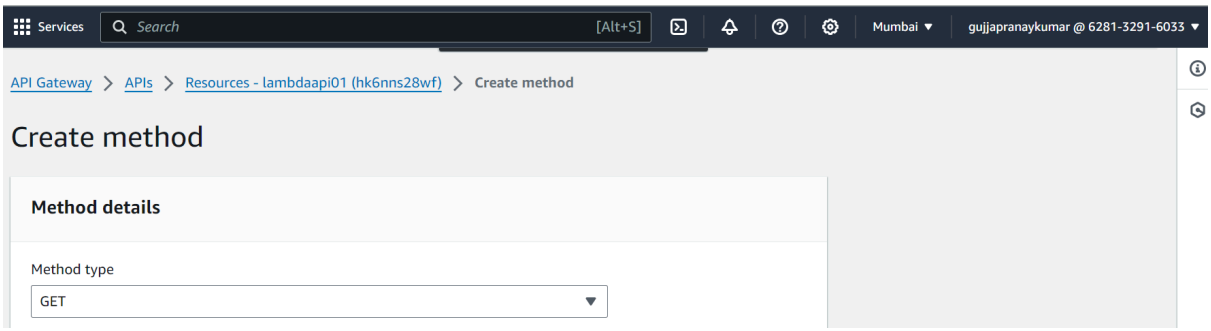
3.Now click on code and deploy it and test it

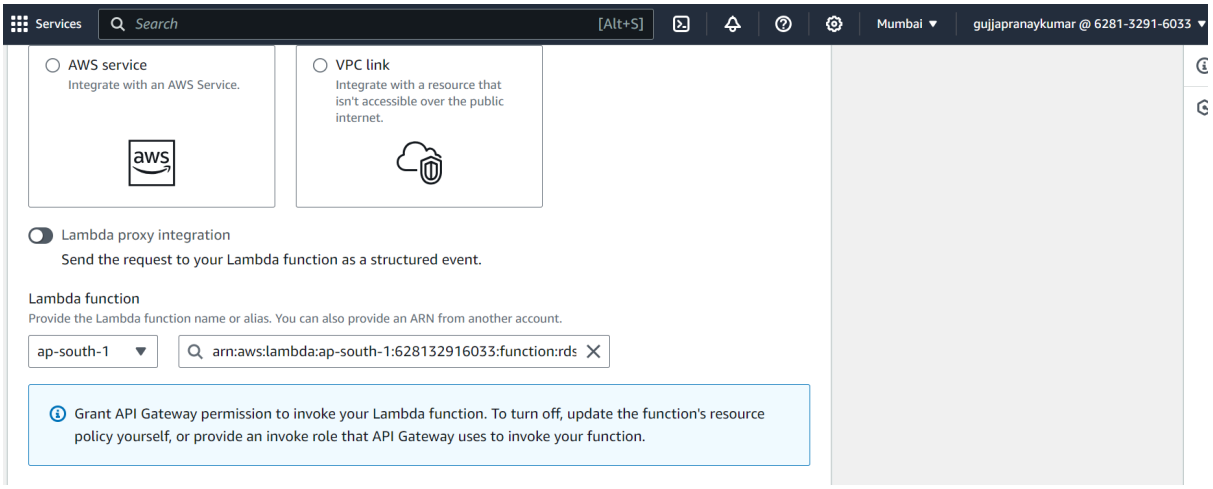
A:

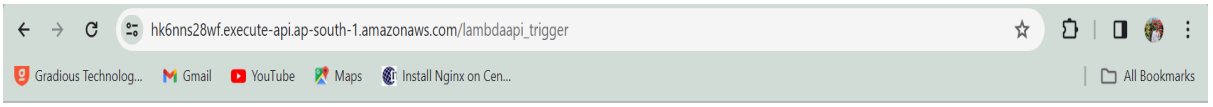


4.You can configure api gateway also to get the response in the browser

A:







{"statusCode":200,"body":"[{\"id\":1,\"name\":\"Ramesh\",\"age\":32,\"address\":\"Ahmedabad\",\"salary\":2000},
{\"id\":2,\"name\":\"Khilan\",\"age\":25,\"address\":\"Delhi\",\"salary\":1500},{\"id\":3,\"name\":\"kaushik\",\"age\":23,\"address\":\"Kota\",\"salary\":2000},
{\"id\":4,\"name\":\"Chaitali\",\"age\":25,\"address\":\"Mumbai\",\"salary\":6500},{\"id\":5,\"name\":\"Hardik\",\"age\":27,\"address\":\"Bhopal\",\"salary\":8500}]"}