Que🡪

Problem Statement: Company ABC wants to move their product to AWS.

They have the following things set up right now:

1. MySQL DB

2. Website (PHP)

The company wants high availability on this product, therefore wants Auto Scaling to be enabled on this website.

Steps To Solve:

1. Launch an EC2 Instance

2. Enable Auto Scaling on these instances (minimum 2)

3. Create an RDS Instance

4. Create Database & Table in RDS instance:

a. Database name: intel

b. Table name: data

c. Database password: intel12345.

Change hostname in website

6. Allow traffic from EC2 to RDS instance

7. Allow all-traffic to EC2 instance

a)So first we need an ec2 instance on which we can run over website.

We have already shared code of php which we are going to use.

First create Ec2 instance.

Click on launch instance.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

We have given name of our ec2 instance and select ubuntu as os for ami.

Select instance type as micro.

A screenshot of a computer

Description automatically generated

Created one key pair.

A screenshot of a chat

Description automatically generated

Let’s create one security group.

A screenshot of a computer

Description automatically generated

We will allow all traffic on http here.

A screenshot of a computer

Description automatically generated

Also add the ssh rule with http.

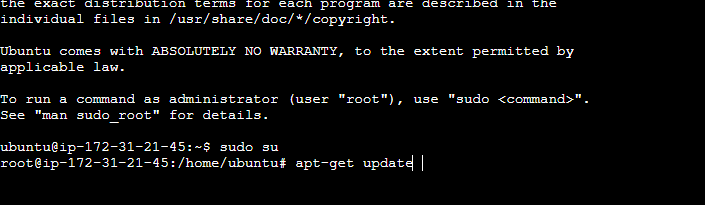
After that launch instance.

A screenshot of a computer

Description automatically generated

Now connect to our instance.

Now first thing is to update our instance.



Now to run our web application we need a server so we are going to install apache server.

A black and white screen with white text

Description automatically generated

Now we can see our apache server running

A screenshot of a computer program

Description automatically generated

By default, Apache web server is instructed to listen for incoming connection and bind on port 80. If you opt for the TLS configuration, the server will listen for secure connections on port 443.

So we have configure our http already so it will listen.

Lets try to open public ip of instance on web page.

A screenshot of a computer

Description automatically generated

We can see apache server healthy and running.

Also we can see the file path to replace for our website.

/var/www/html/index.html.

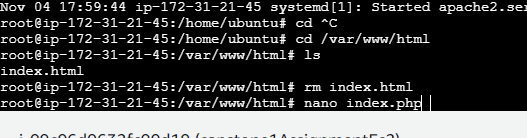
Let’s go to this path and replace it with our php code.

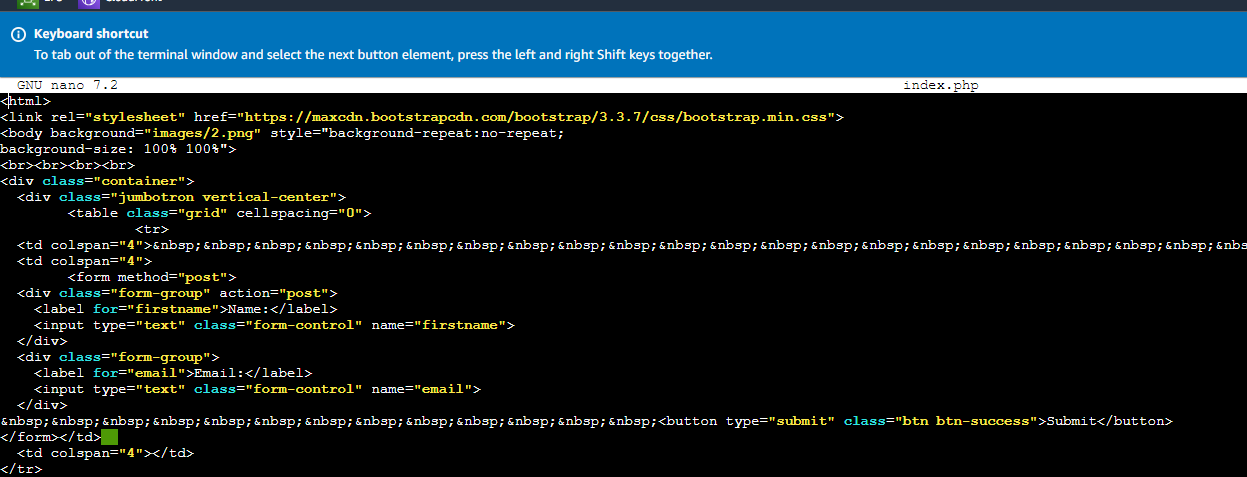
A computer screen with white text

Description automatically generated

First remove file using rm index.html

Than use nano index.php and copy paste our code.





Now if you refresh the page

A screenshot of a computer

Description automatically generated

We can see our web page loading with some errors , why?

Because it is not able to read php , so we need to install php compiler into our instance.

A screen shot of a computer

Description automatically generated

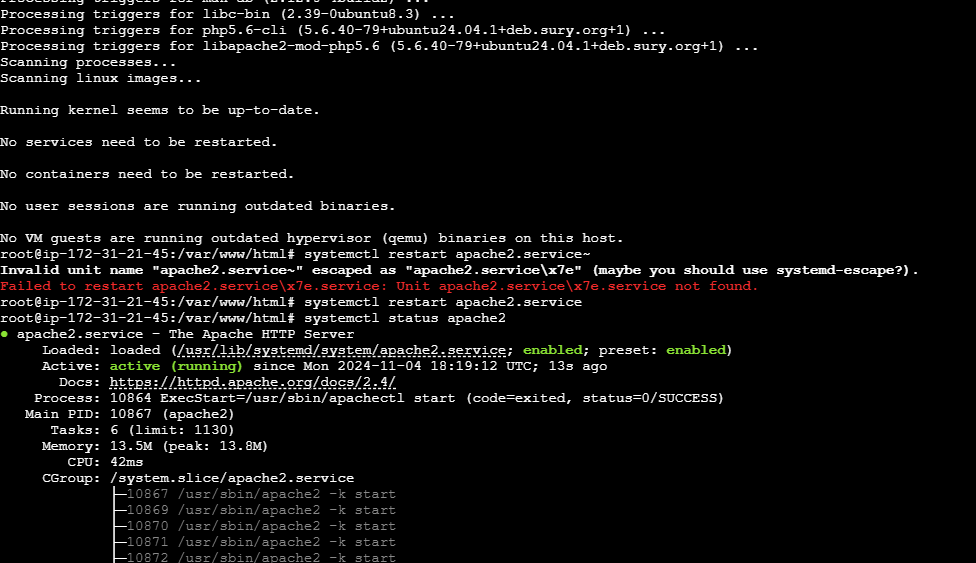
Now to connect php with mysql we need other client like maria db.

A screen shot of a computer

Description automatically generated

Now restart apache server using command

sudo systemctl restart apache2.service



Now refresh browser.

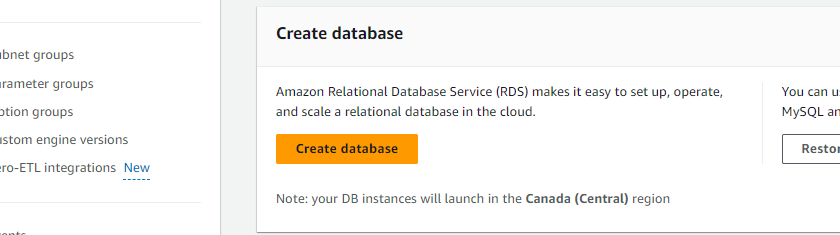
Error gone only connection left.

A screenshot of a computer

Description automatically generated

Now let’s create a rds instance.

Click on create database.



Select mysql.

A screenshot of a computer

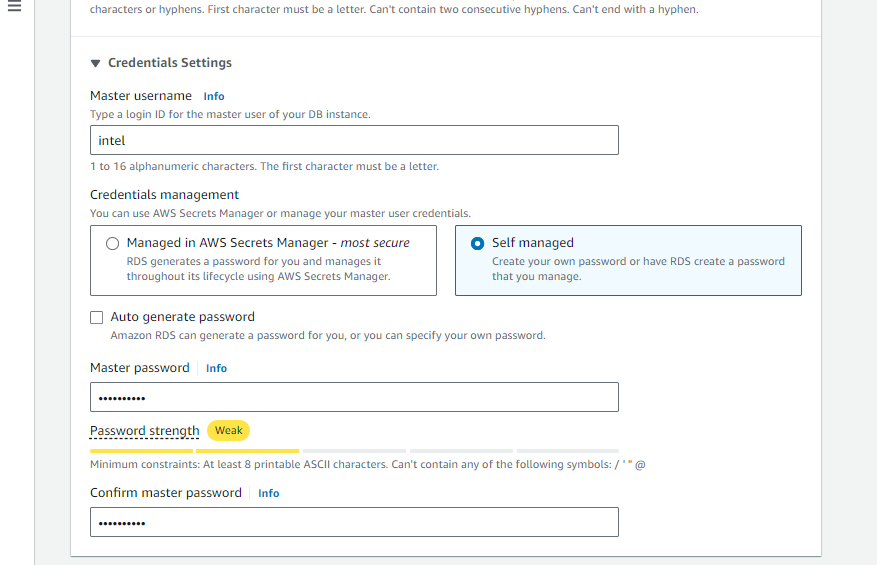
Description automatically generated

Let’s use free tier for know.

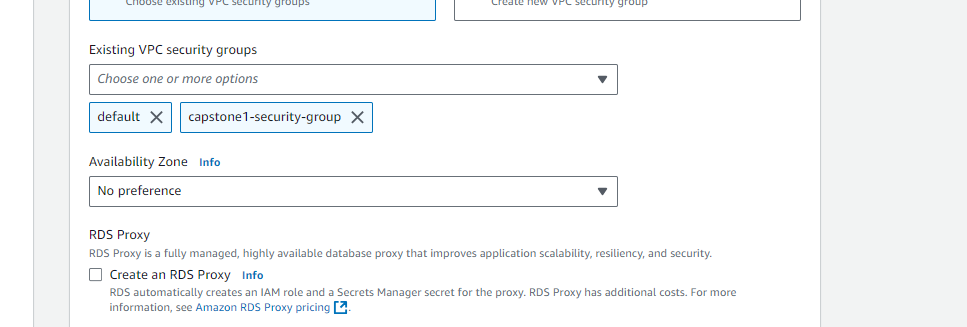
A screenshot of a computer

Description automatically generated

We have create a user with name intel and password intel12345.



Choose our security group



Give intel name to our database . left other setting as it is.

A screenshot of a computer

Description automatically generated

Create database.

To connect Rds with our Ec2 instance we have to edit inbound rules for security group to get a successful connection.

A screenshot of a computer

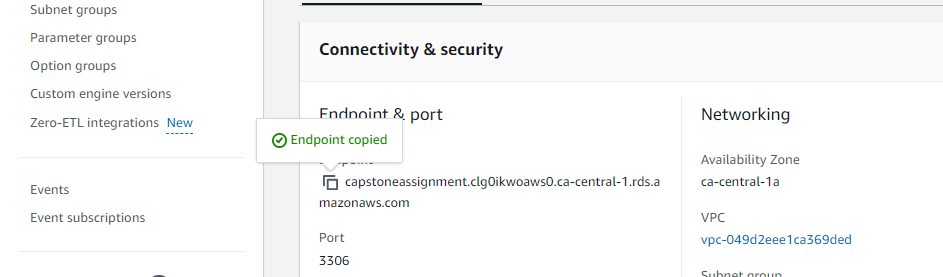
Description automatically generated

We have added mysql/aurora in inbound rules.

Now try to connect ec2 instance with our rds using command.

mysql -h capstoneassignment.clg0ikwoaws0.ca-central-1.rds.amazonaws.com -u intel -p

here -h is url of our database which you can find here.



Now we have connected to our database.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer screen

Description automatically generated

Now create a table in database with name data and two properties firstname and email.

A screenshot of a computer

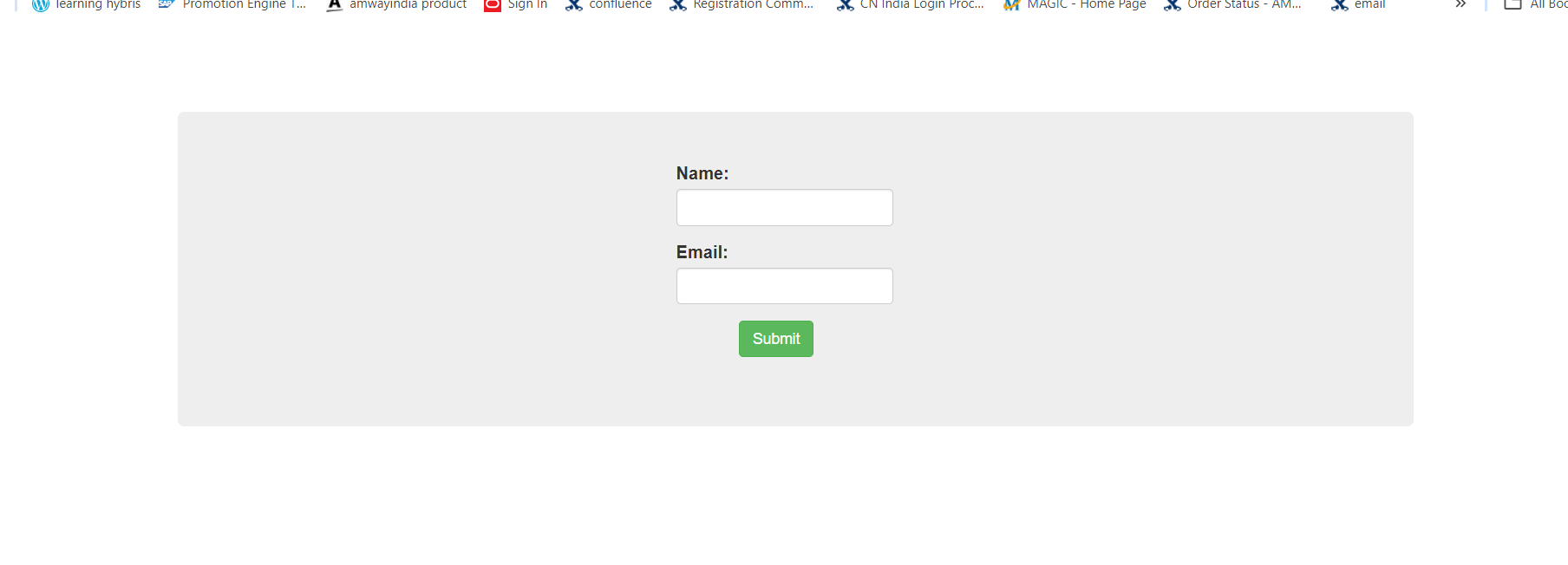
Description automatically generated

We have created and inserted one value into the table . now edit our index.php



Change severname / password whatever you have set and save it.

Now you can see no error in web page of connection.



Now try to submit some value.

A screenshot of a computer

Description automatically generated

After submitting the button go into your data base and see all entries in table.

We can see new values.

Now our ec2 part is done now the left one is auto scaling.

So first let’s create image of our ec2 instance which we are going to use in launch template for our auto scaling group.

A screenshot of a computer

Description automatically generated

Let’s create image

A screenshot of a computer

Description automatically generated

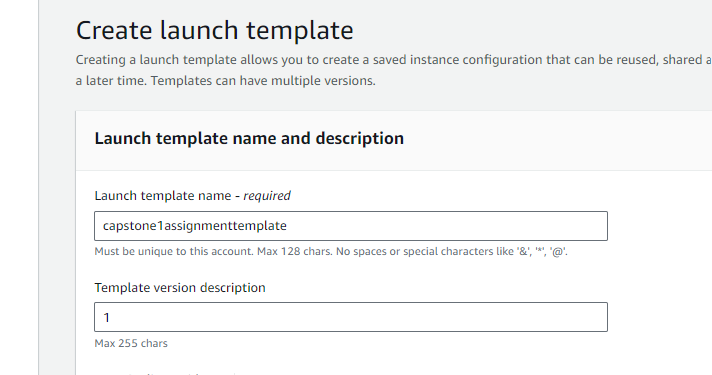
After successful creation of image delete your instance.

A screenshot of a computer

Description automatically generated

Now our ami is available.

Now let’s create a launch template.



In AMI’s select our AMI.

A screenshot of a computer

Description automatically generated

Select your key-pair and security group.

A screenshot of a computer

Description automatically generated

Create launch template.

A screenshot of a computer

Description automatically generated

Now go and create Auto Scaling Group.

A screenshot of a computer

Description automatically generated

Select our launch template.

A screenshot of a computer

Description automatically generated

Choose network setting.

A screenshot of a computer

Description automatically generated

Let’s Attach a load balancer for our application to distribute load.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

As we need minimum 2 instances so in desired we have select

A screenshot of a computer

Description automatically generated

Skip all and create auto scaling groups.

In activities we can see our instances is launched.

A screenshot of a computer

Description automatically generated

If you go into instances we can see running instances of ours.

A screenshot of a computer

Description automatically generated

Now go to our load balancer.

A screenshot of a computer

Description automatically generated

Try to connect on web using dns.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Now our assignment is successfully completed.