**Name:** Gourav Kashyap

**Project name**: Deploying a multi-tier website using EC2 aws

**Description**:

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing

capacity in the Amazon Web Services (AWS) cloud. Using Amazon EC2

eliminates your need to invest in hardware up front so you can develop and

deploy applications faster. You can use Amazon EC2 to launch as many or as

few virtual servers as you need, configure security and networking, and manage

storage. Amazon EC2 enables you to scale up or down to handle changes in

requirements or spikes in popularity, reducing your need to forecast traffic.

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

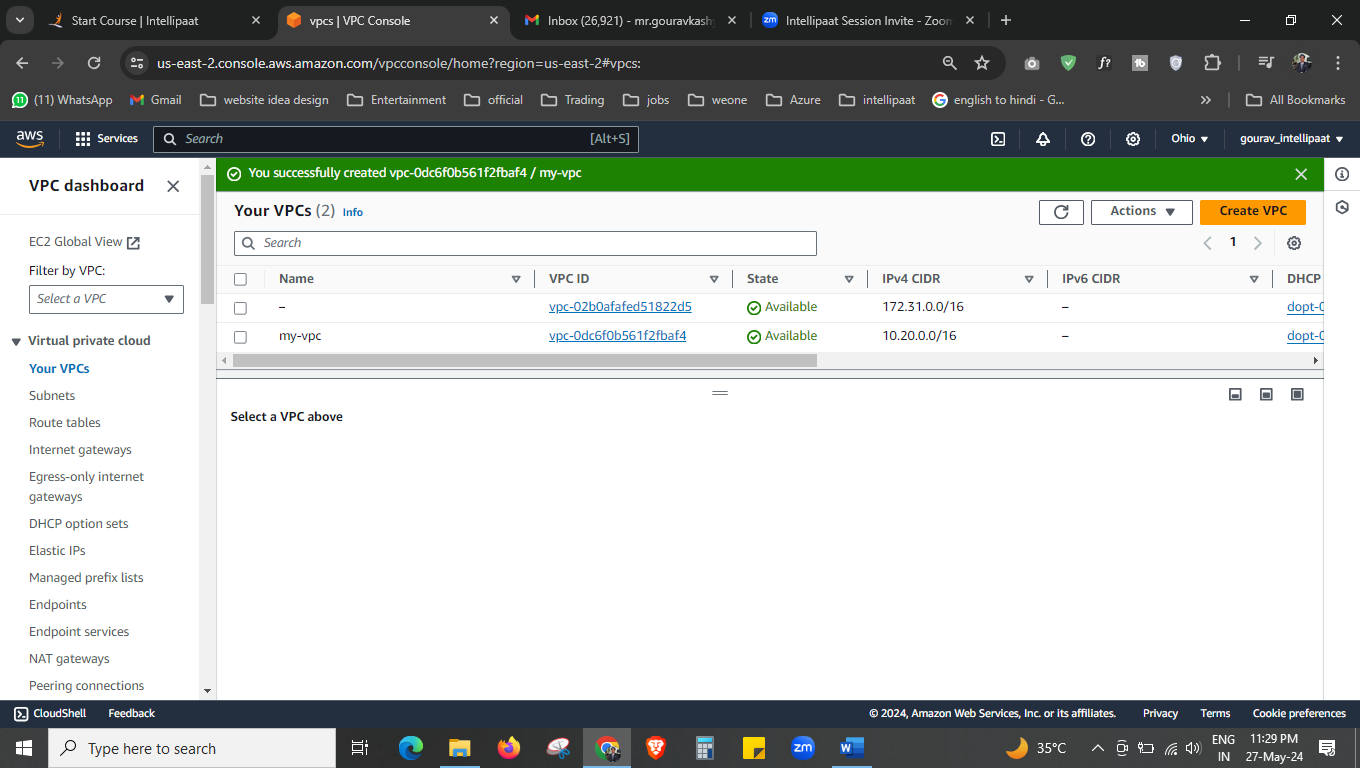
b. Table name: data

c. Database password: intel123

5. Change hostname in website

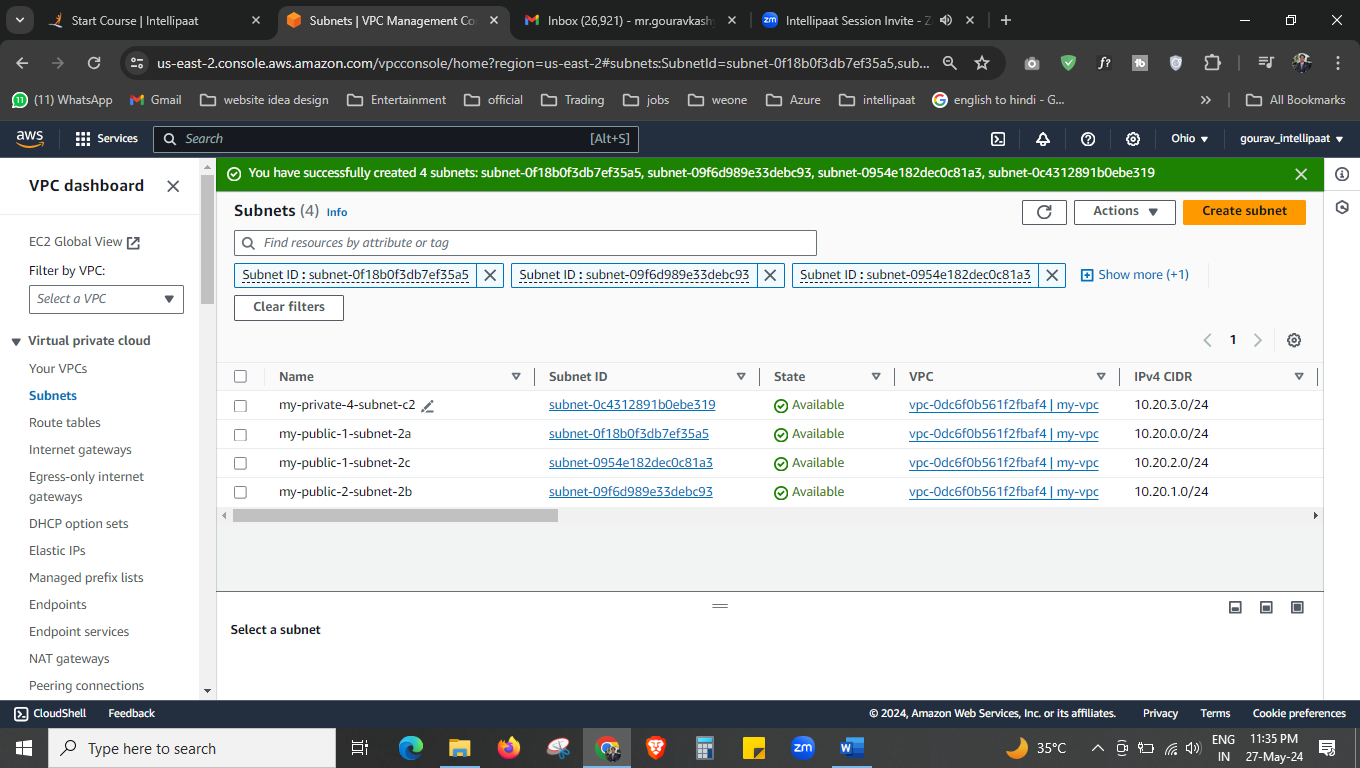
6. Allow traffic from EC2 to RDS instance

7. Allow all-traffic to EC2 instance





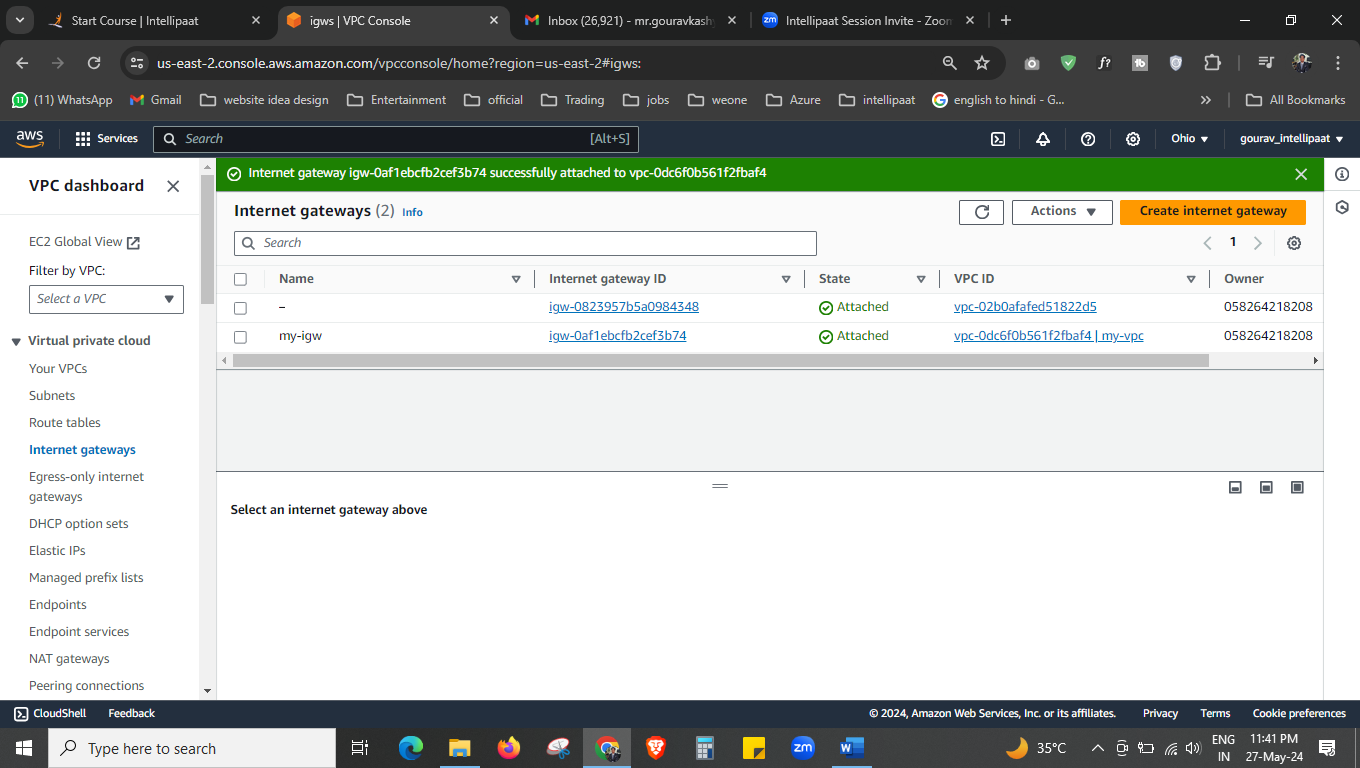
First, we have created the VPC called “my-vpc”





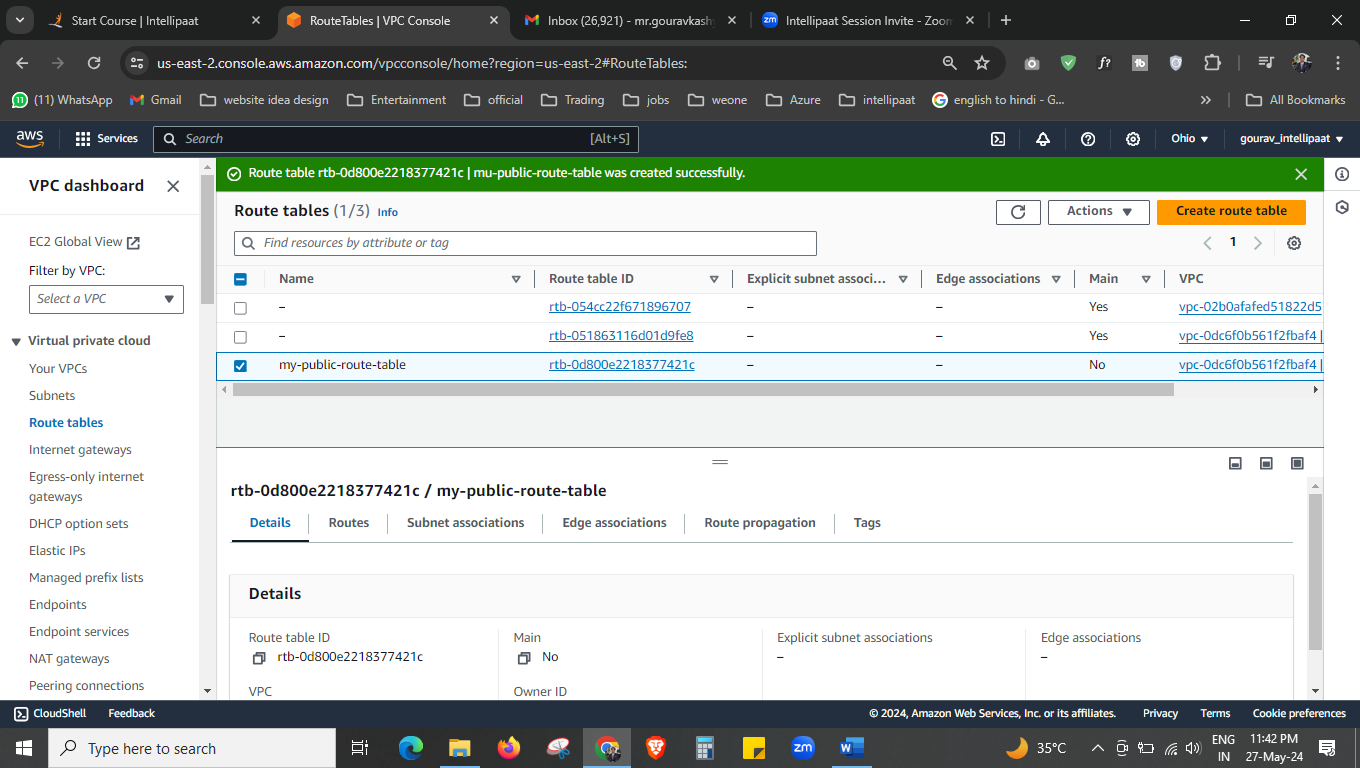
We have created 4 subnets.

3 public subnet and 1 private subnet.





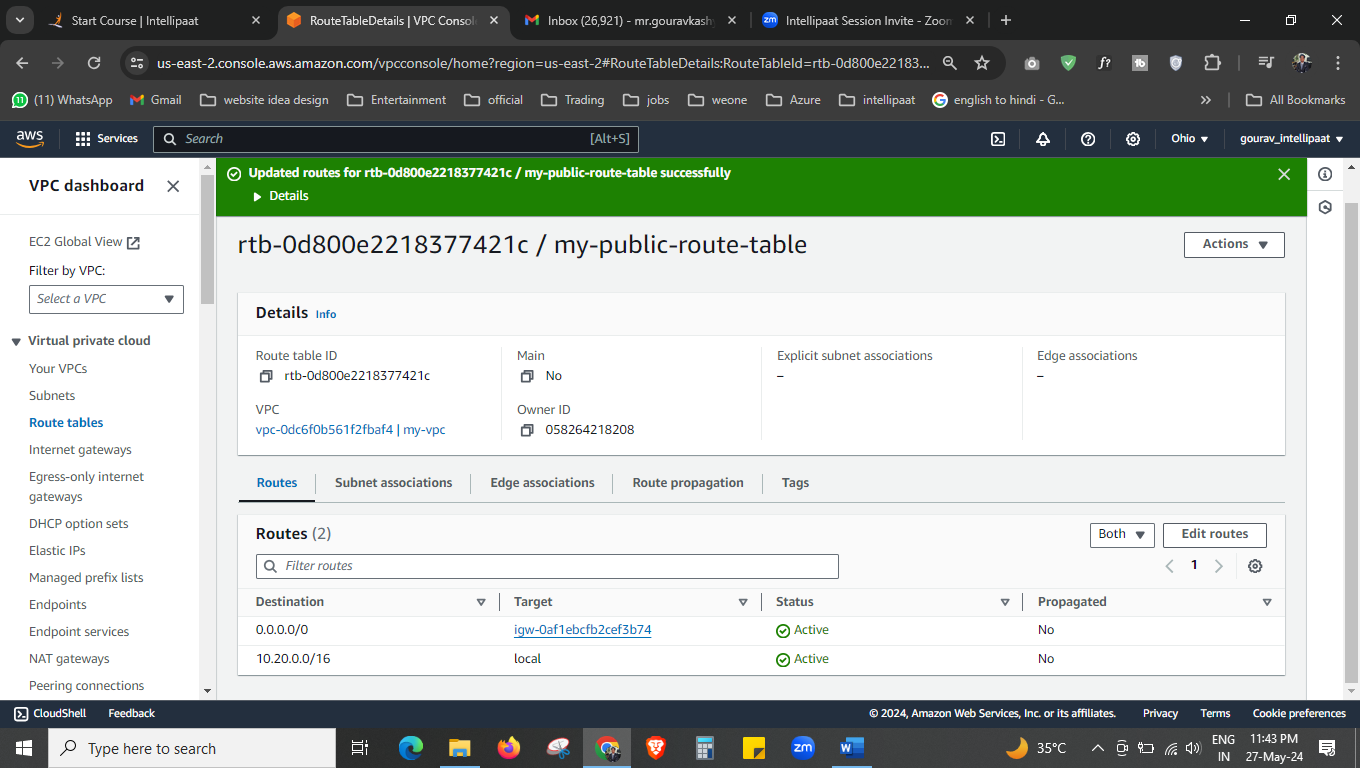
We created the internet gateway now to make route table public.





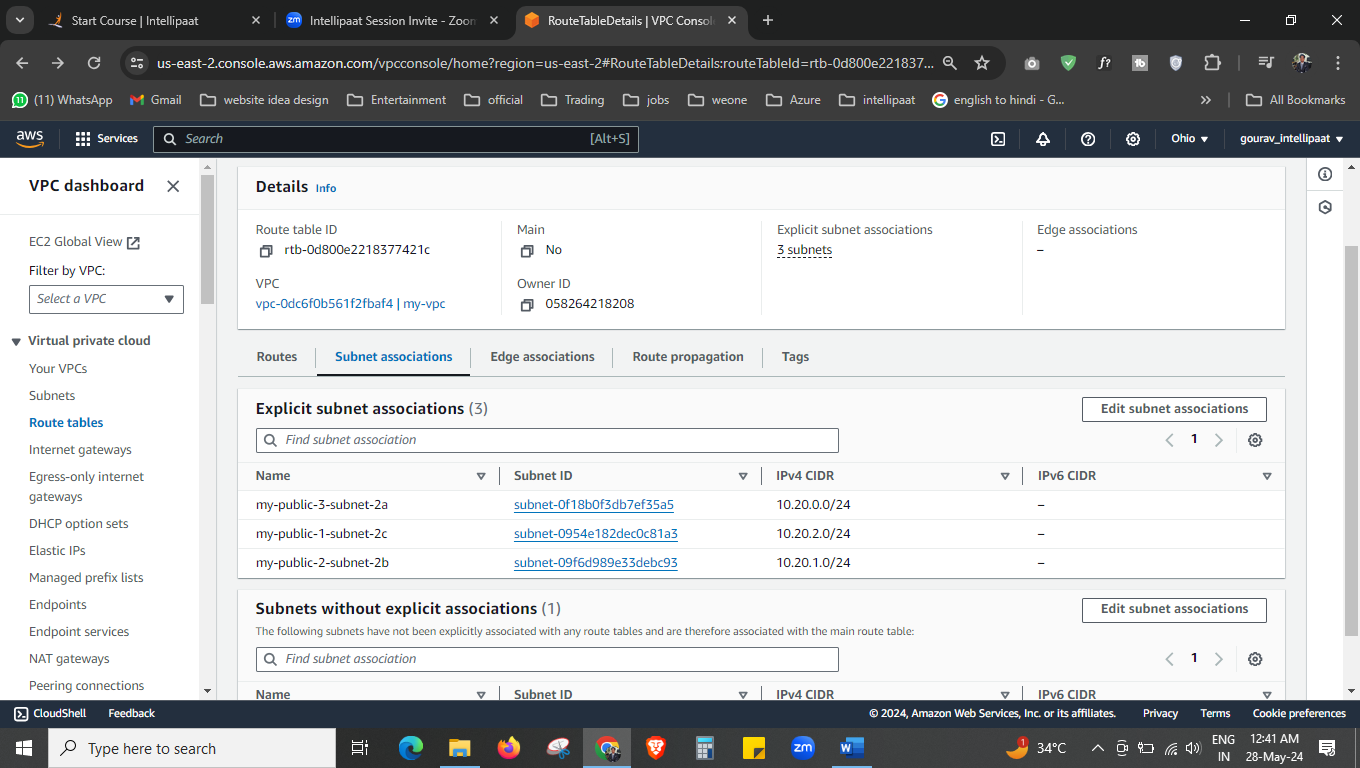
Create route table, then





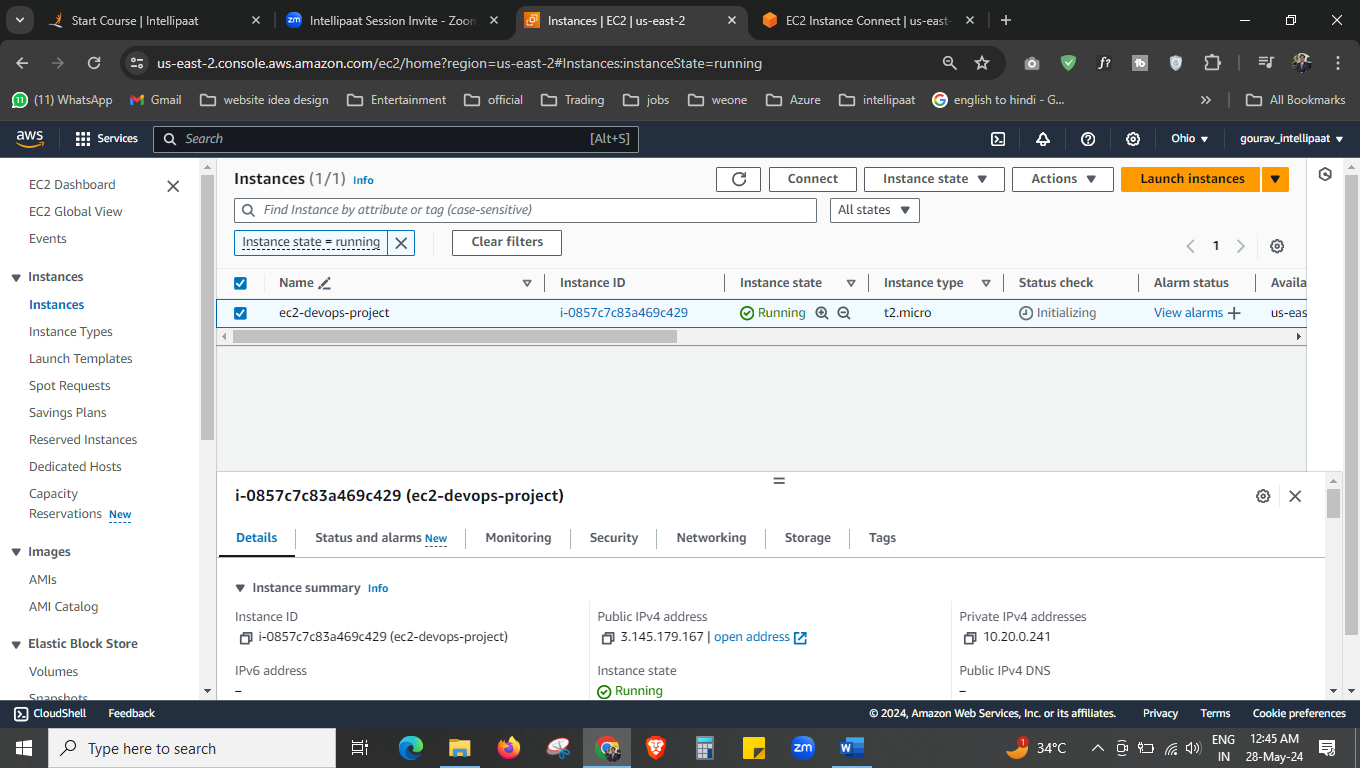


Now to the route table we have associate internet gateway.

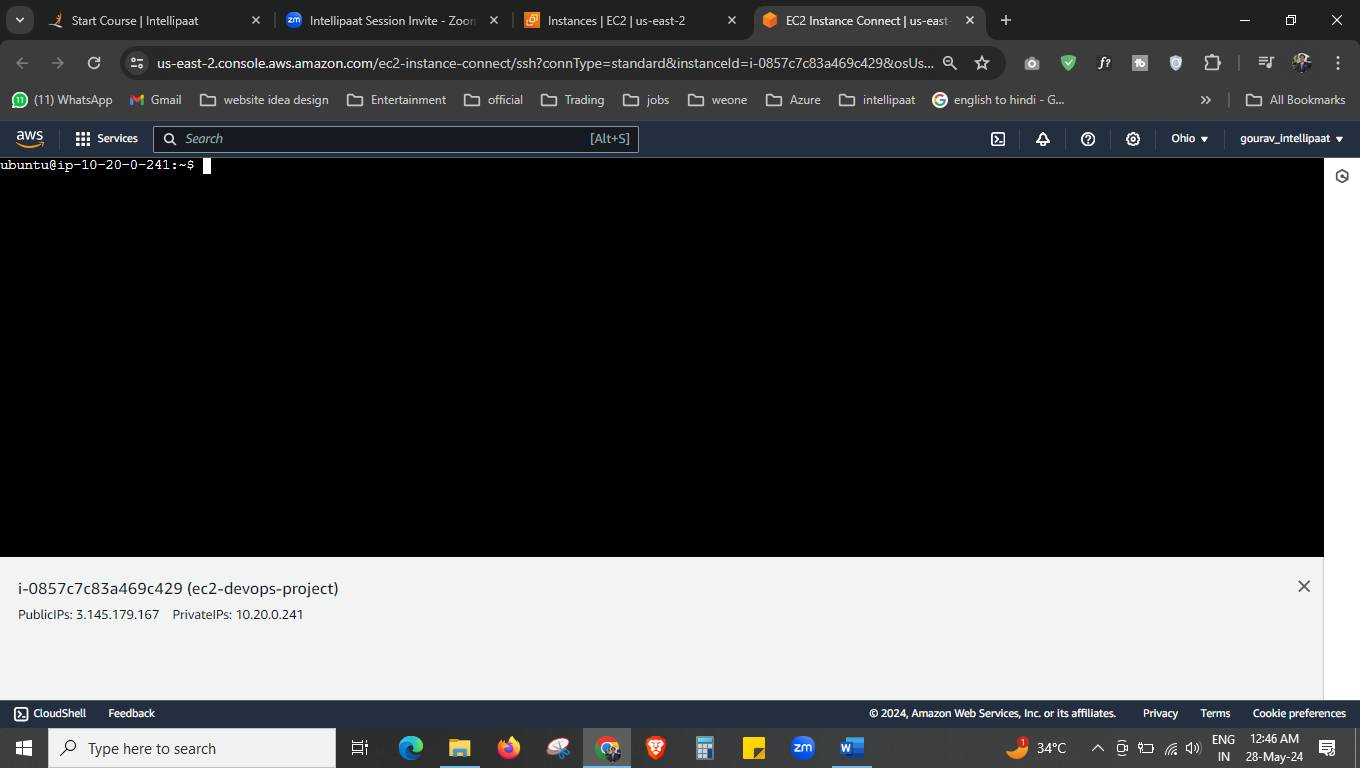
.



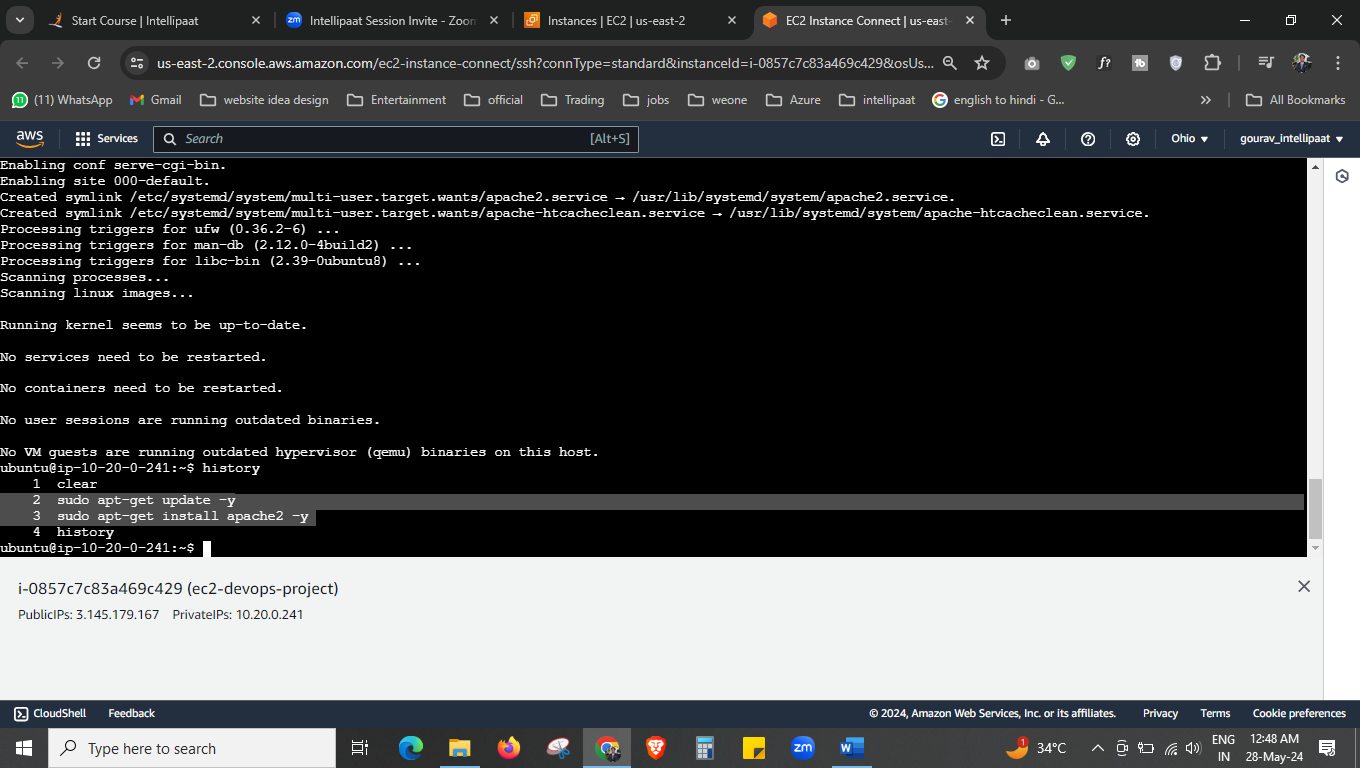
Now associate all 3 public subnets to the route table.



Now we have created the ec2 instance, in our VPC in public-subnet-3



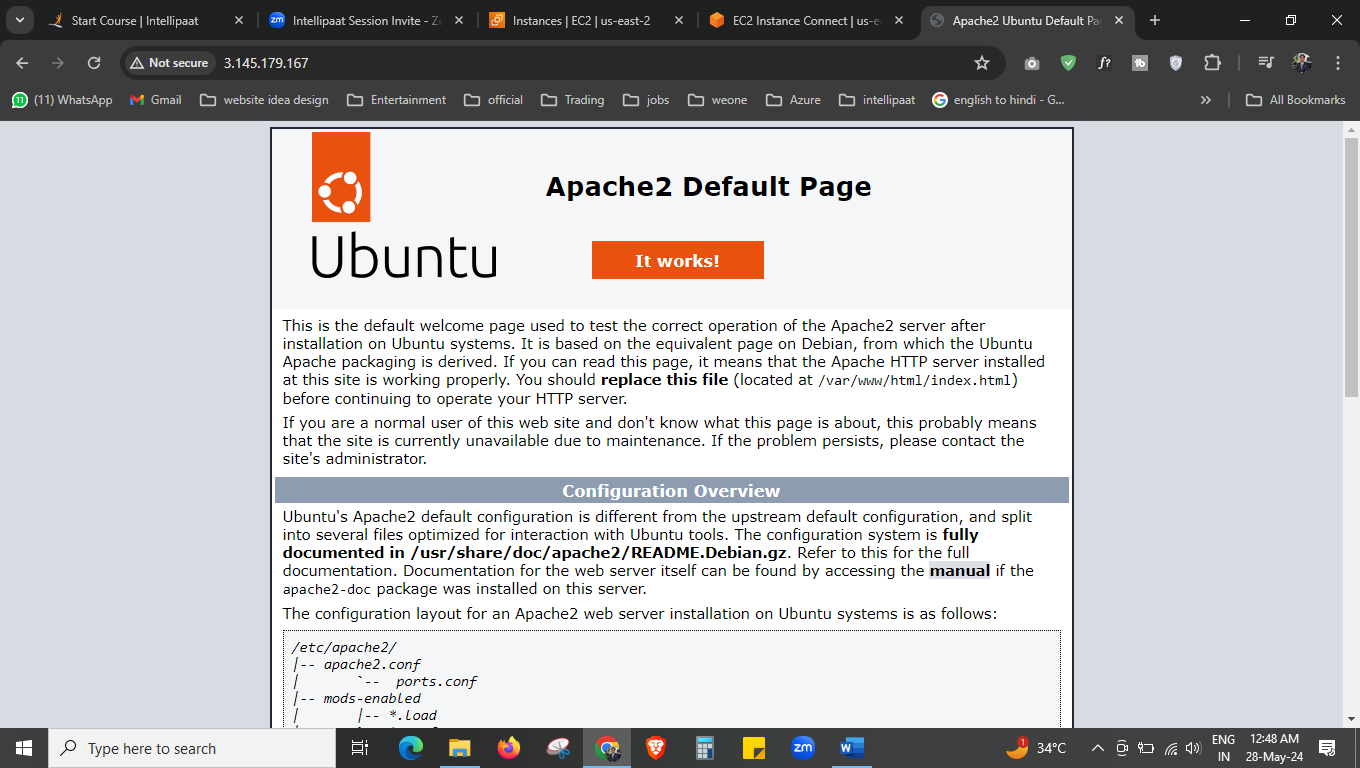
Connection successful.



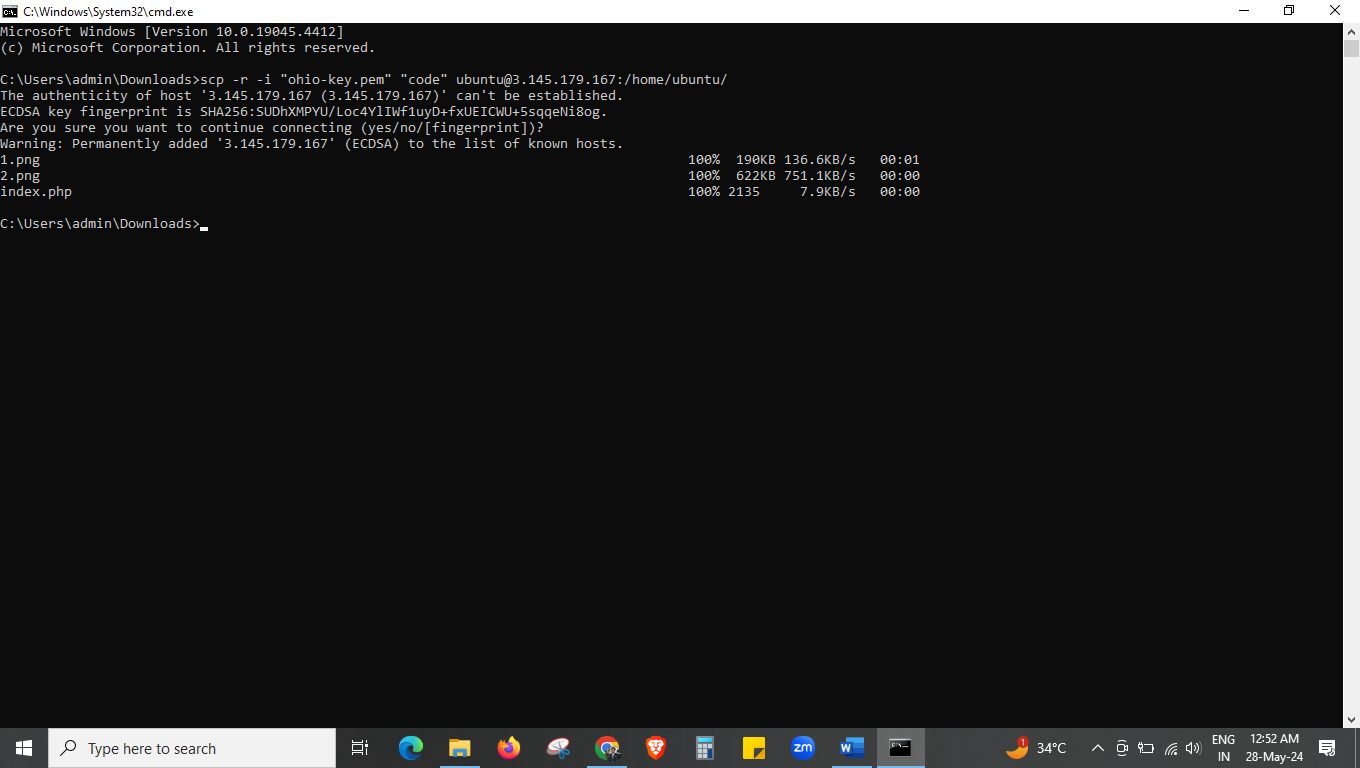
We have installed apache2 in our ec2 instance,

sudo apt-get update -y

sudo apt-get install apache2 -y

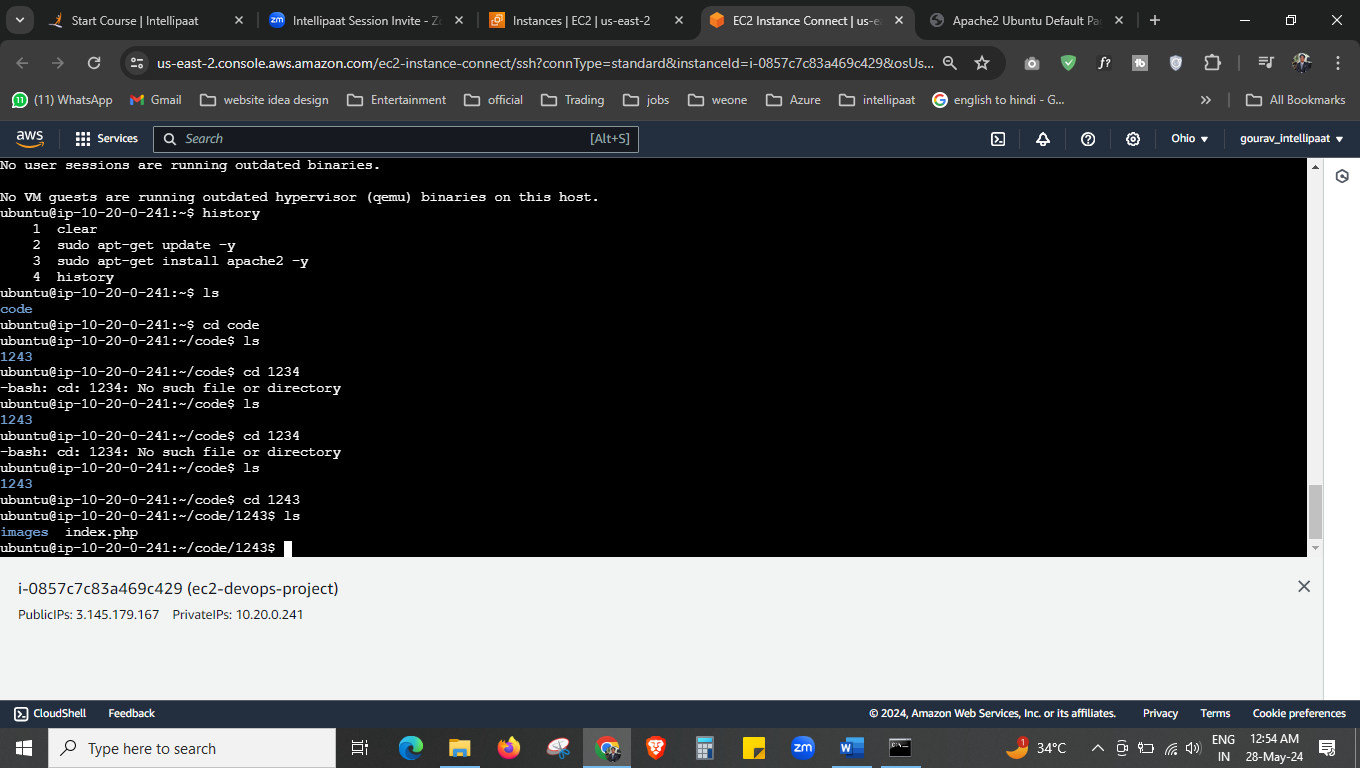


This is the output.



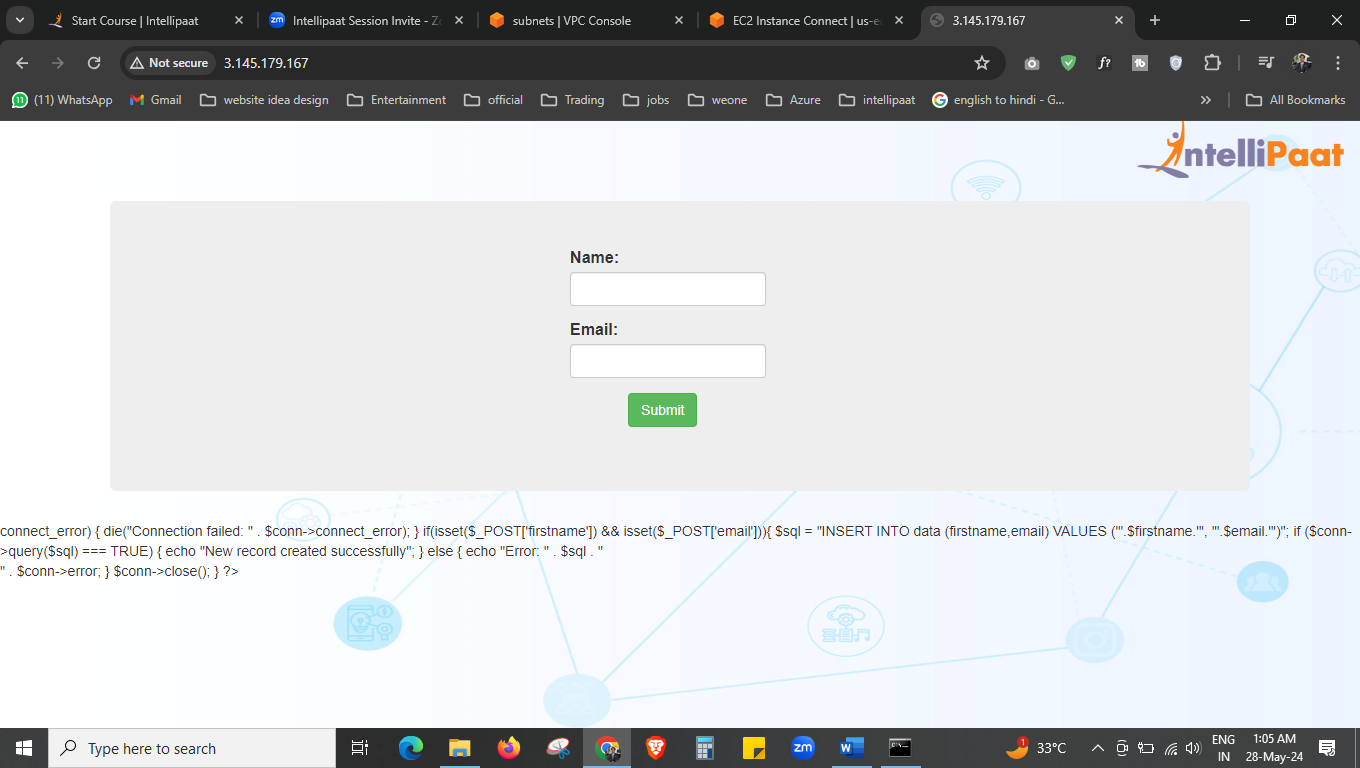
Now we have transferred the website code folder to our instance.

Using scp -r -i “ohio-key.pem” “code” ubuntu@<public-ip>:/home/ubuntu



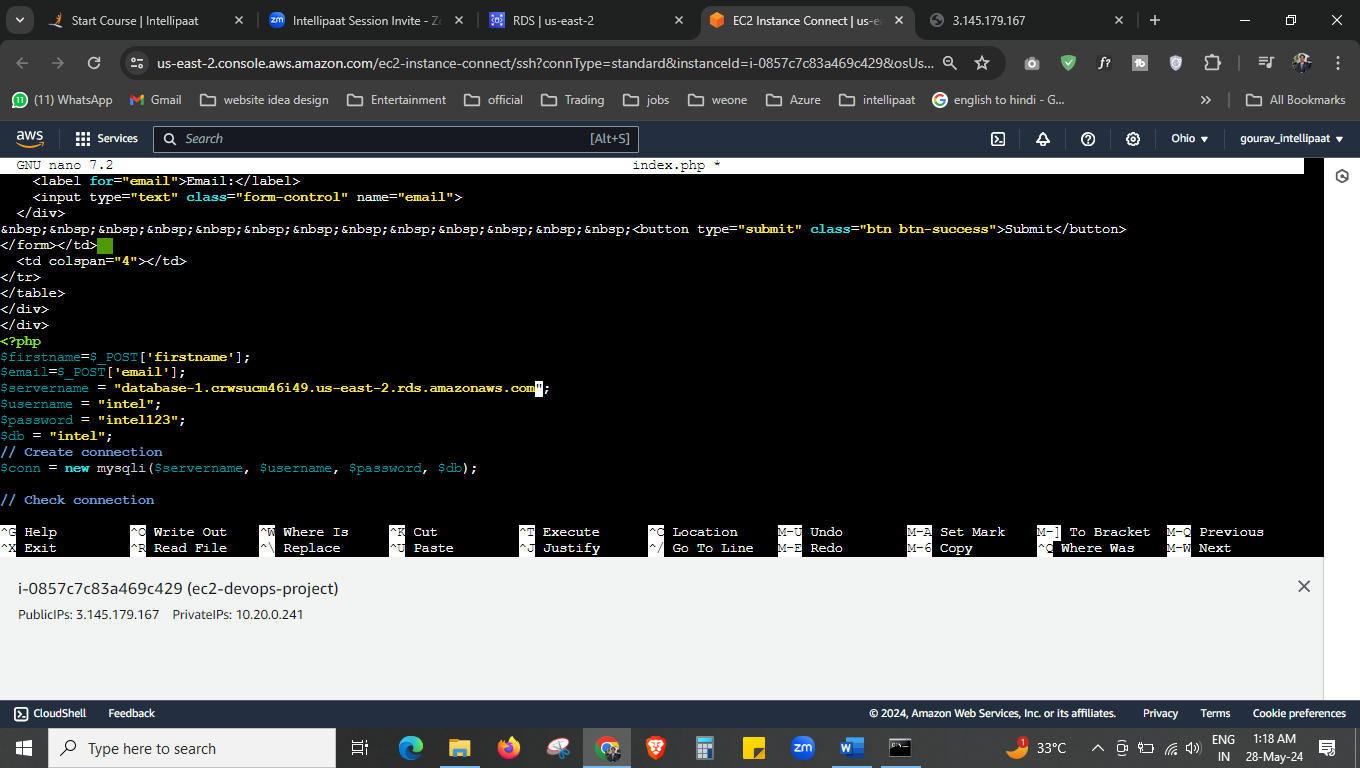


Now you can have these files, which we just transferred from local to ec2.



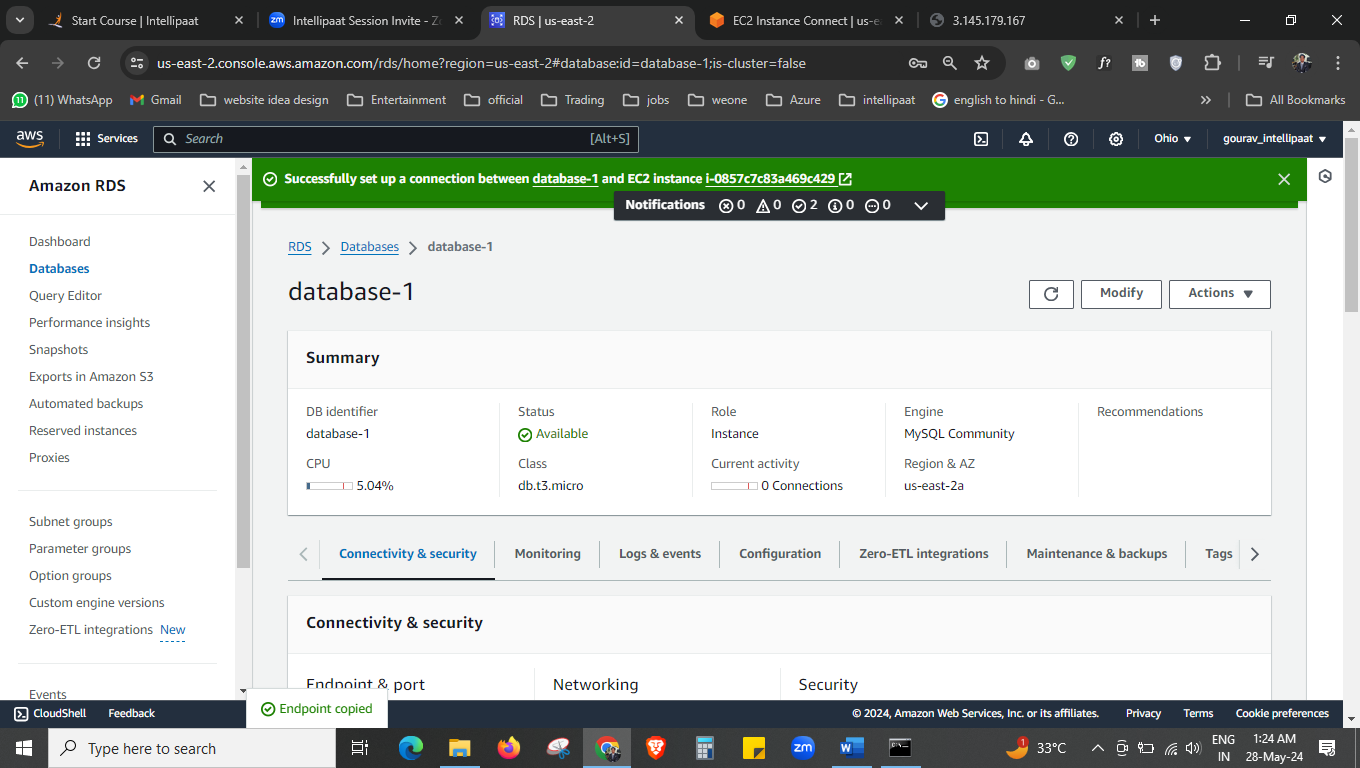
We set our index.php file in /var/www/html and now we have this output

But the form will not work for now. For that we will create the MySQL database now.



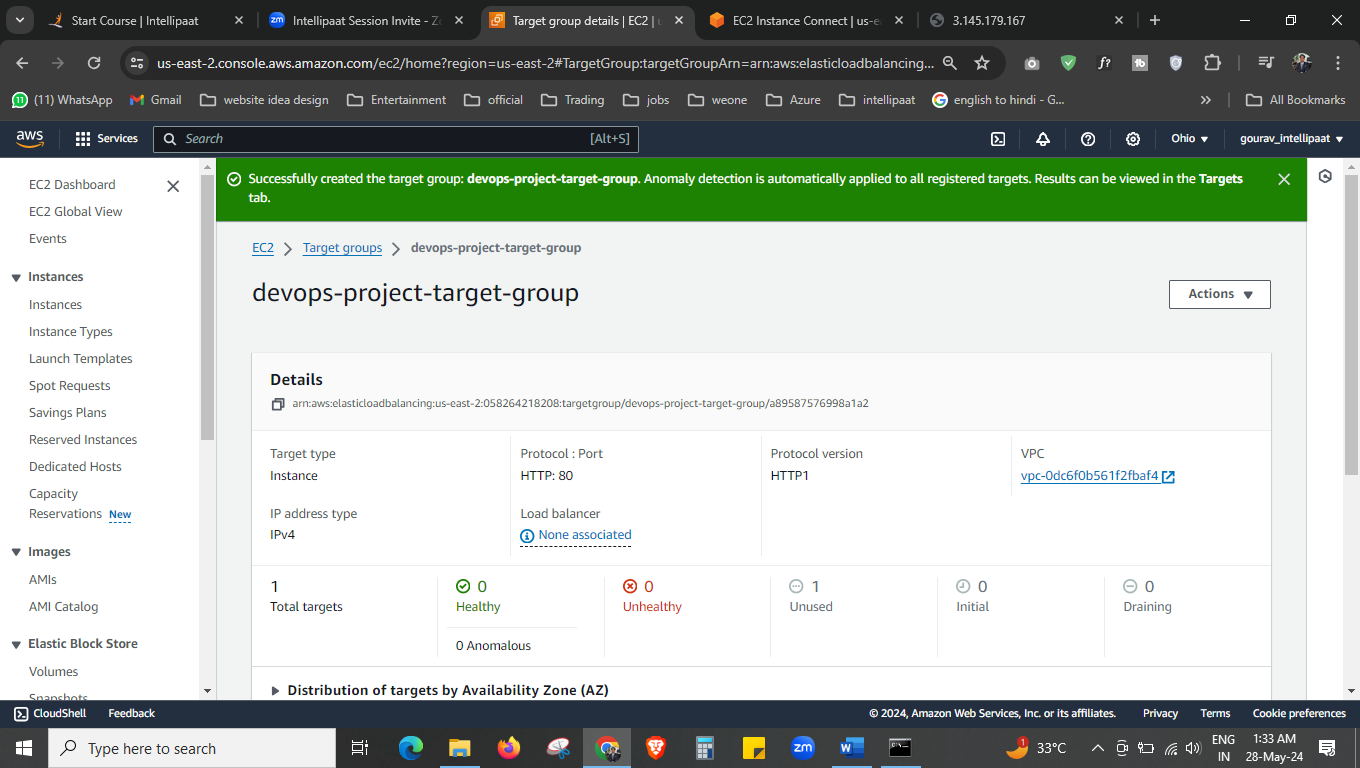


We have added the endpoint of our database to our php code.



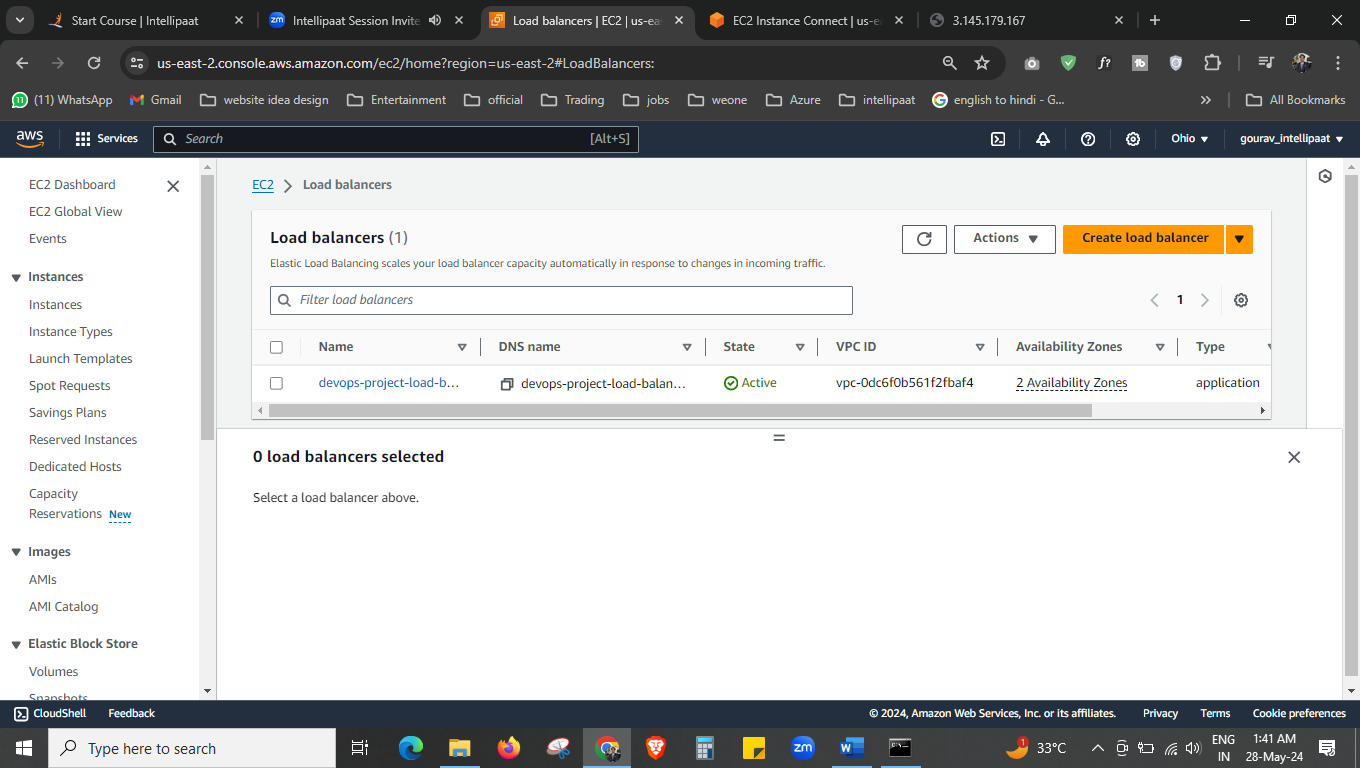
Our database is created.

I: e. mysql



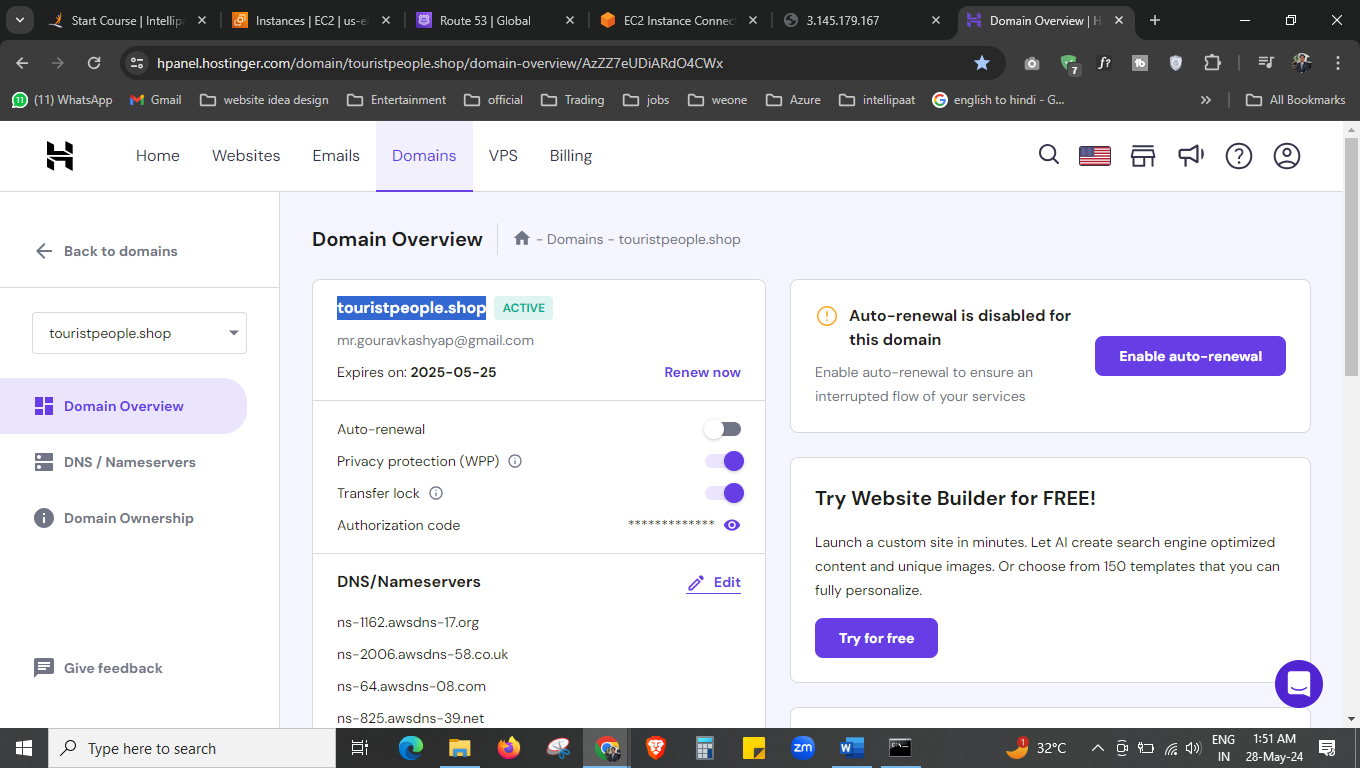


Now we have to create the load balancer but first to create the load balancer we have to create the target-group, here we have created.



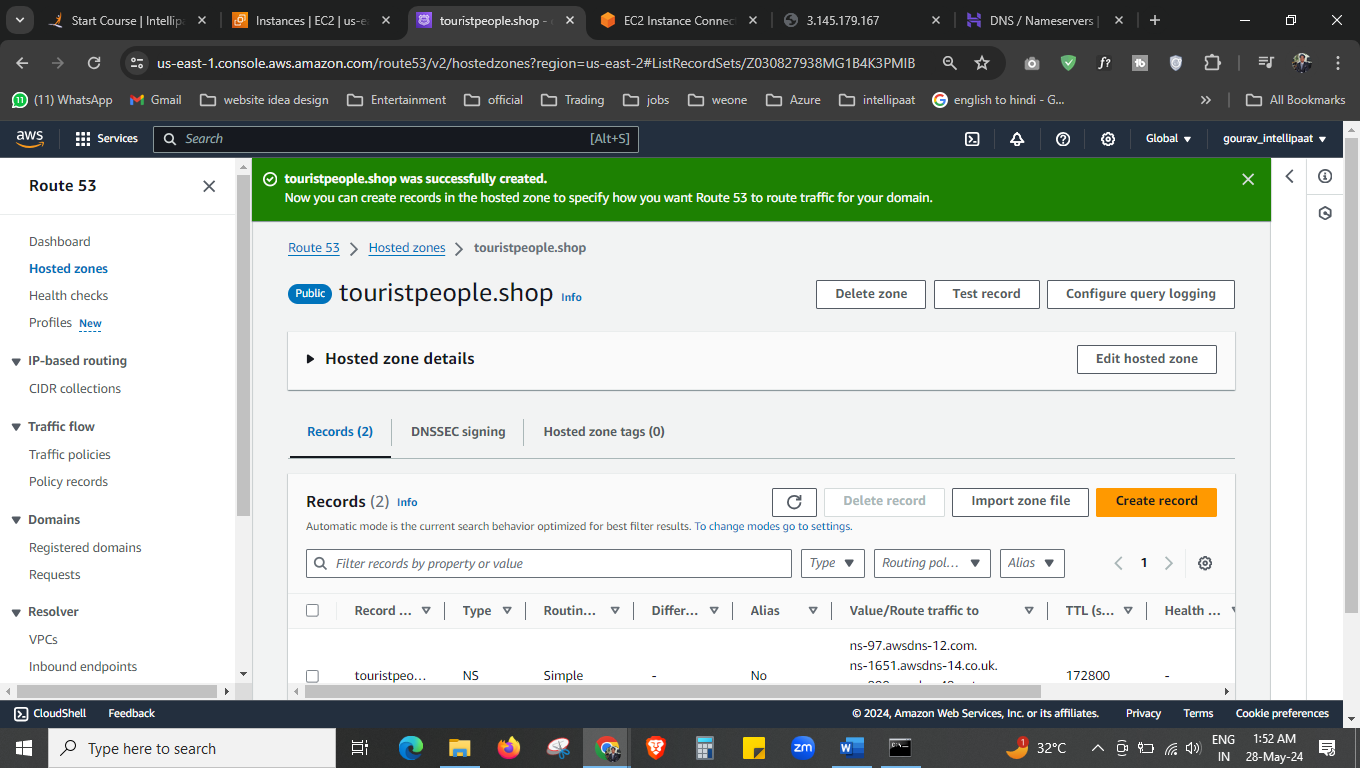


Load balancer is ready and running now

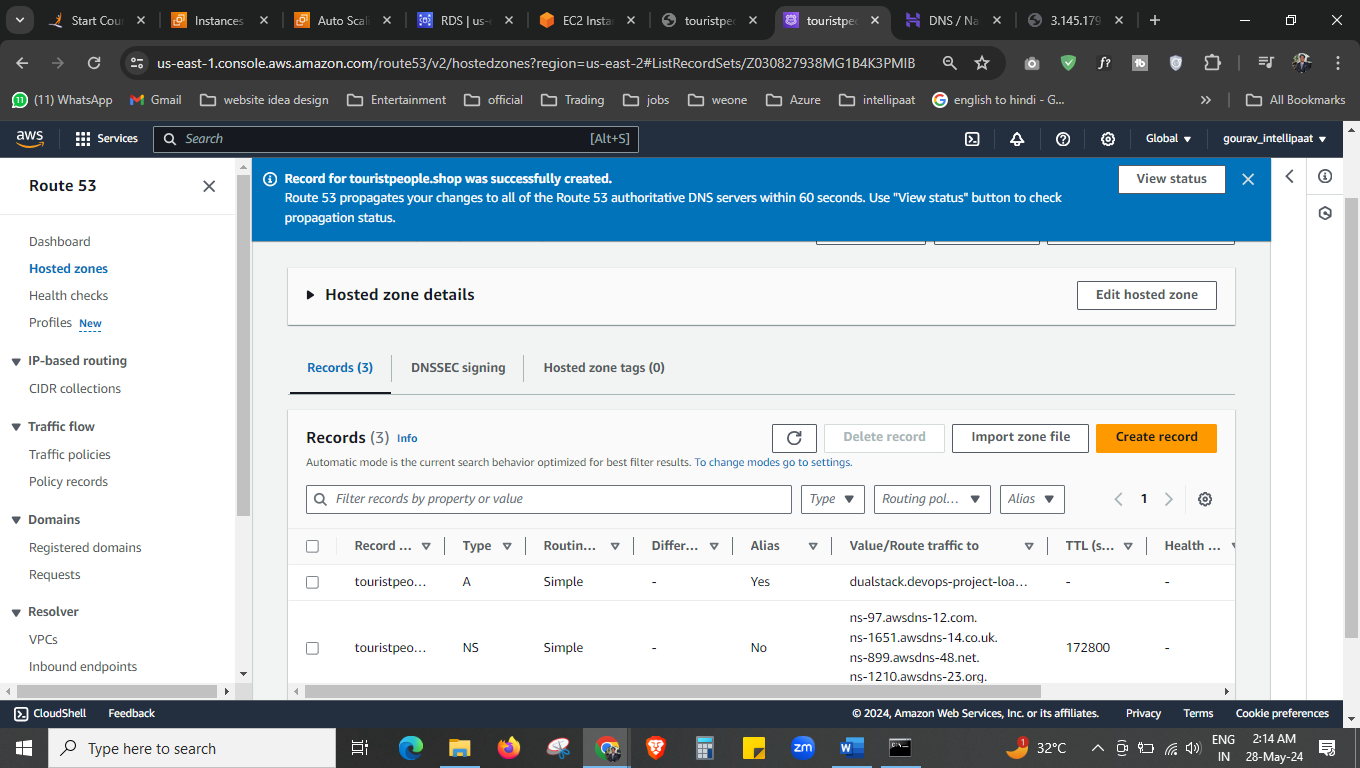




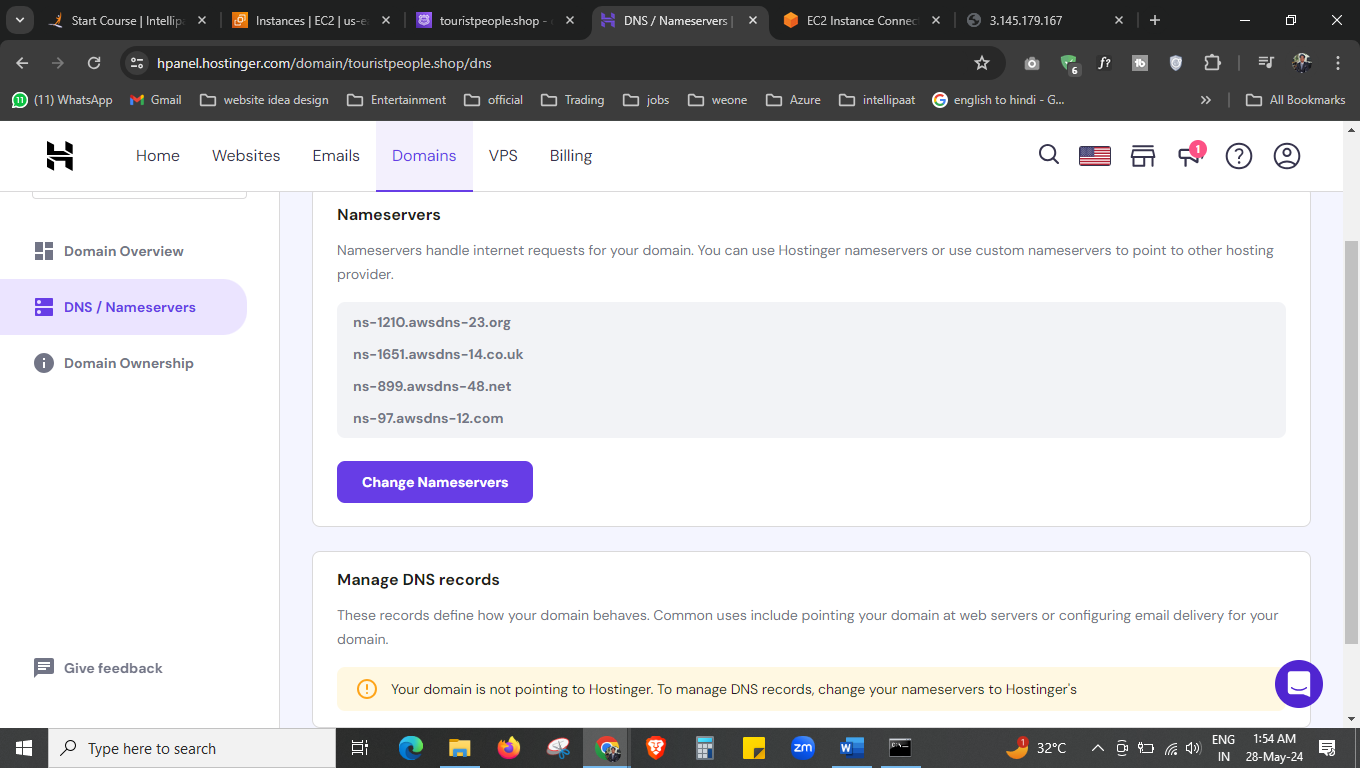
This is our domain we have purchased from hostinger.



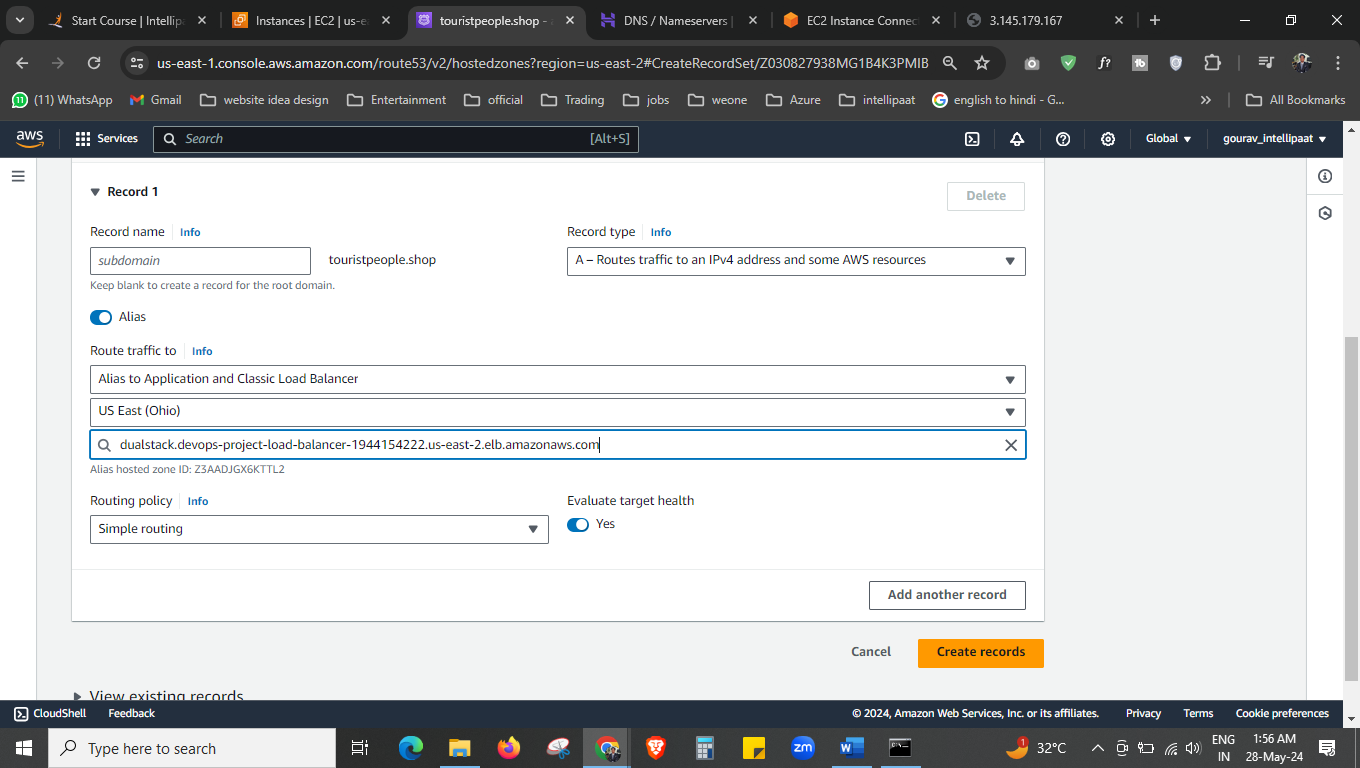
Route 53 we have created now.



Route 53



Now we have changed our name server

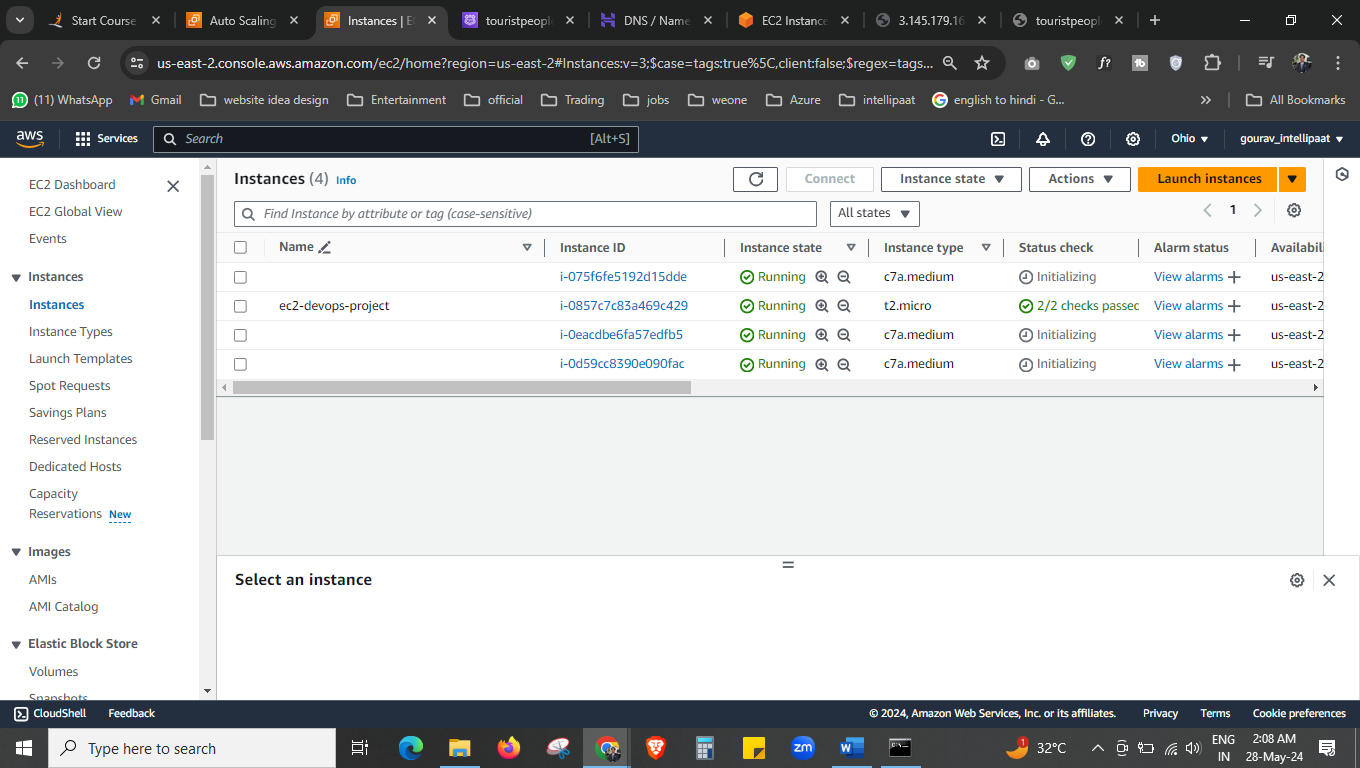




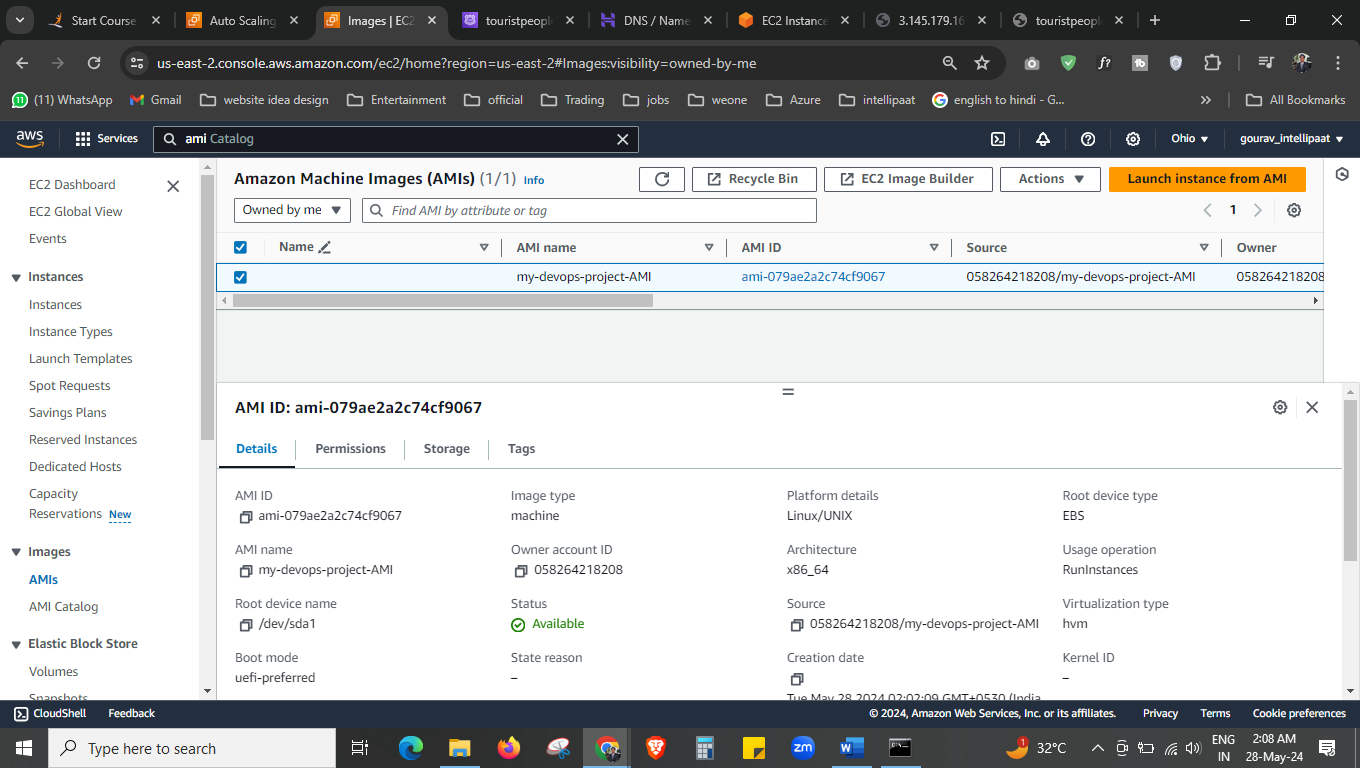
We have to create the record now using alias.



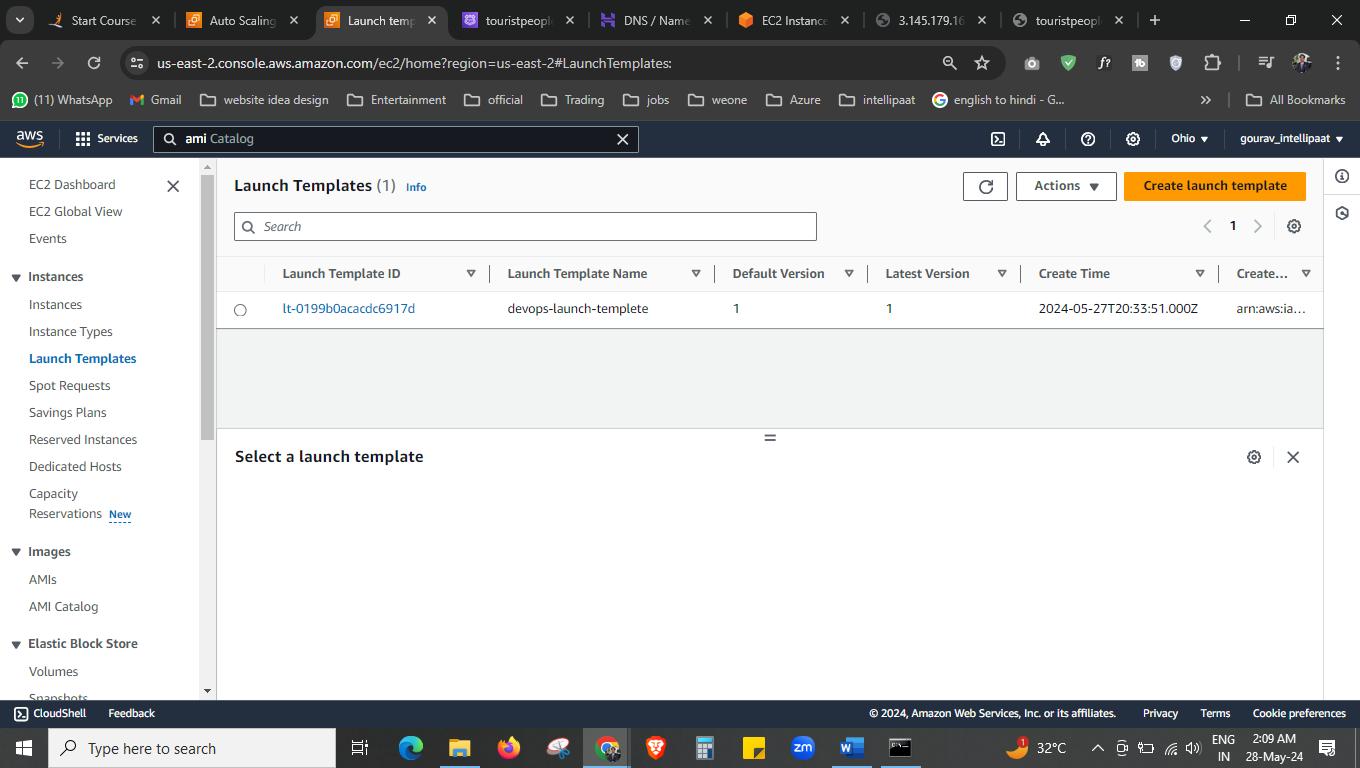
Auto scaling we have launched



Desired ec2 instance is 3 so its launched now.



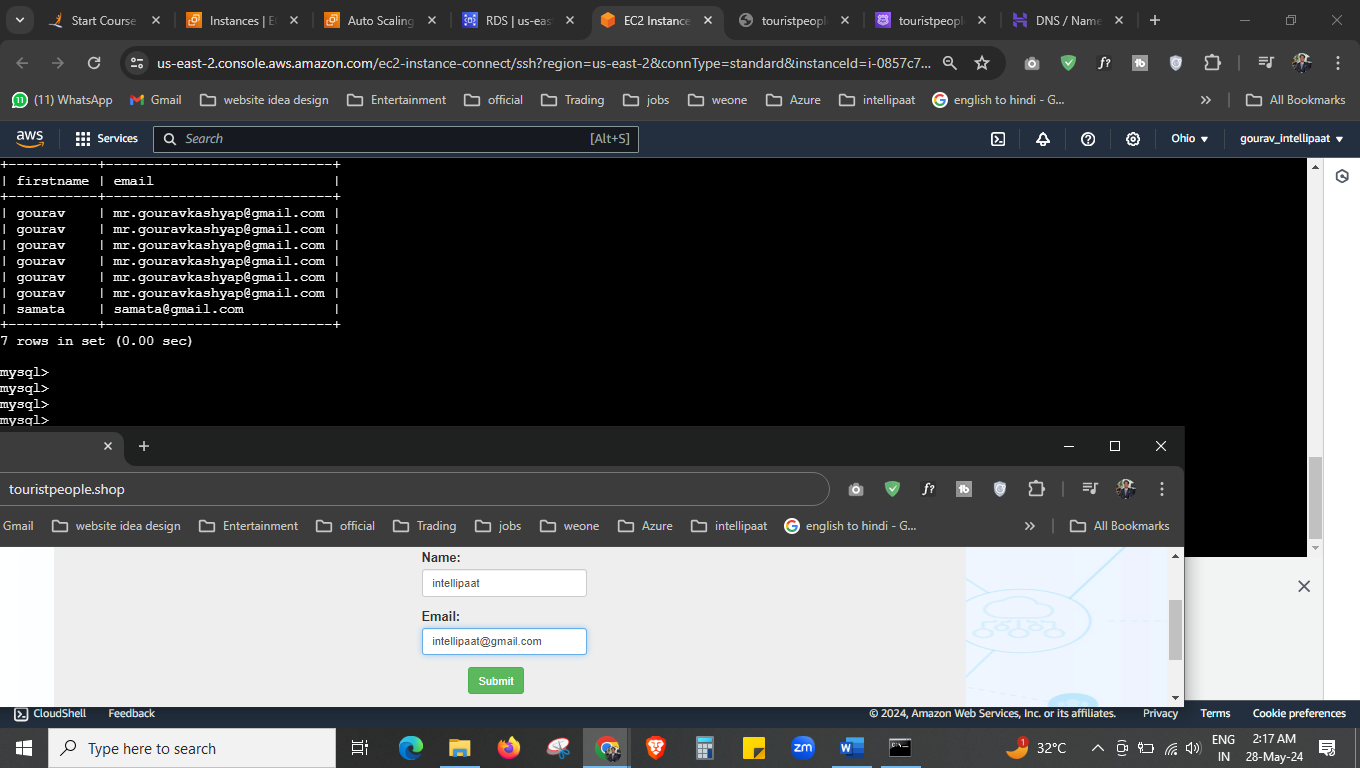
We have AMI also.

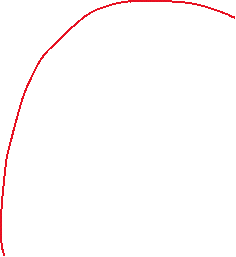
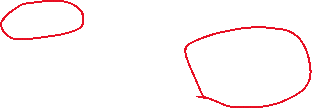


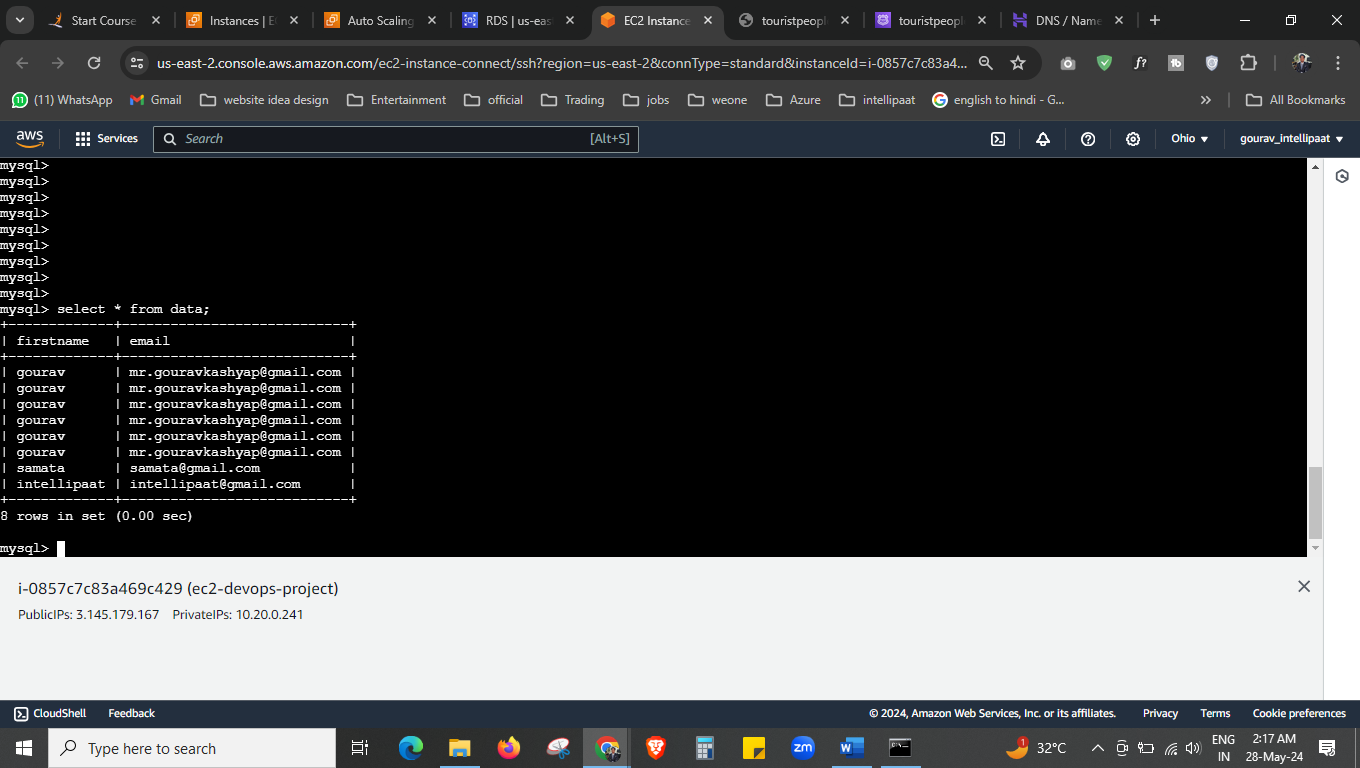


For auto scaling we have launch template also.

Everything is created now.









Done!!! Working website with storing data.