

# Project - 1: Deploying a Multi-Tier Website Using AWS EC2

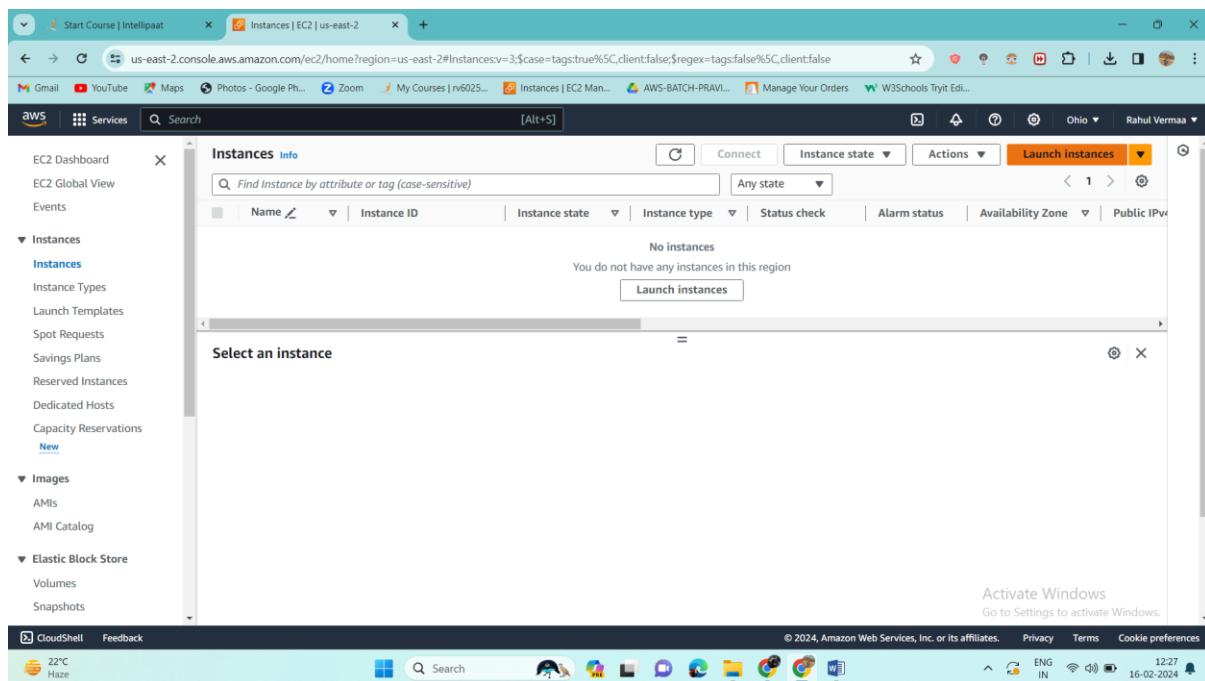
**Description:** Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud. Using Amazon EC2 eliminates your need to invest in hardware up front so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as 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.

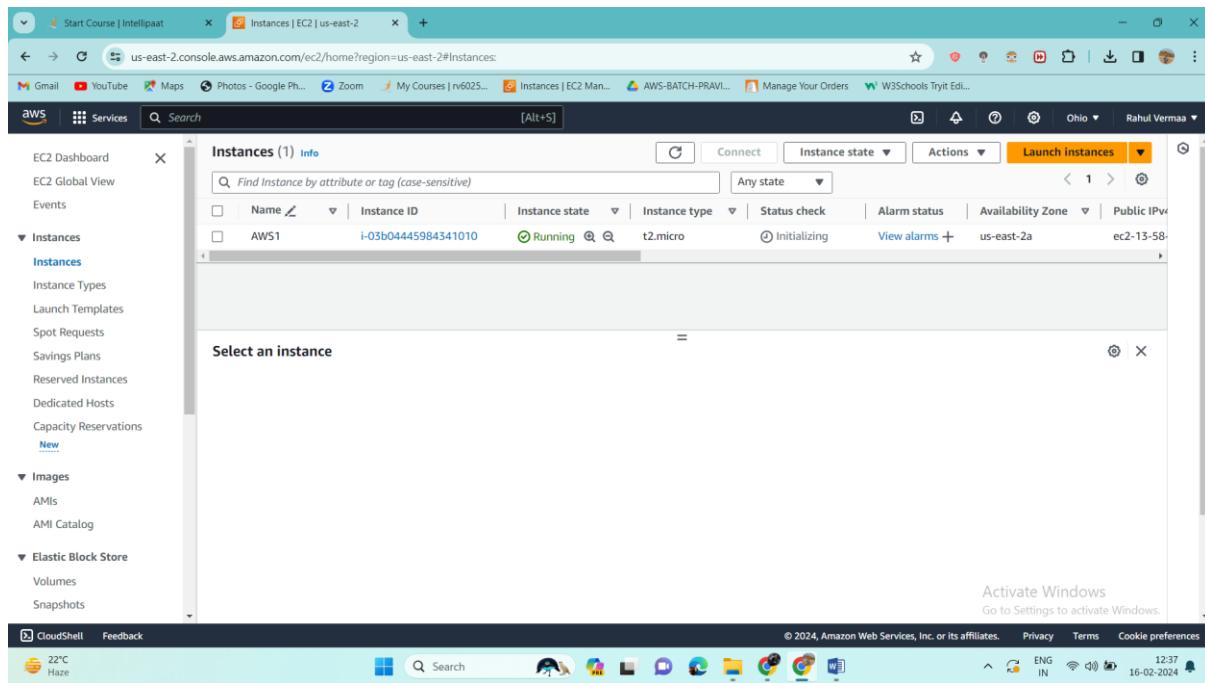
## Steps To Solve:

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 host name in website
6. Allow traffic from EC2 to RDS instance
7. Allow all traffic to EC2 instance.

## Open AWS console and create an instance

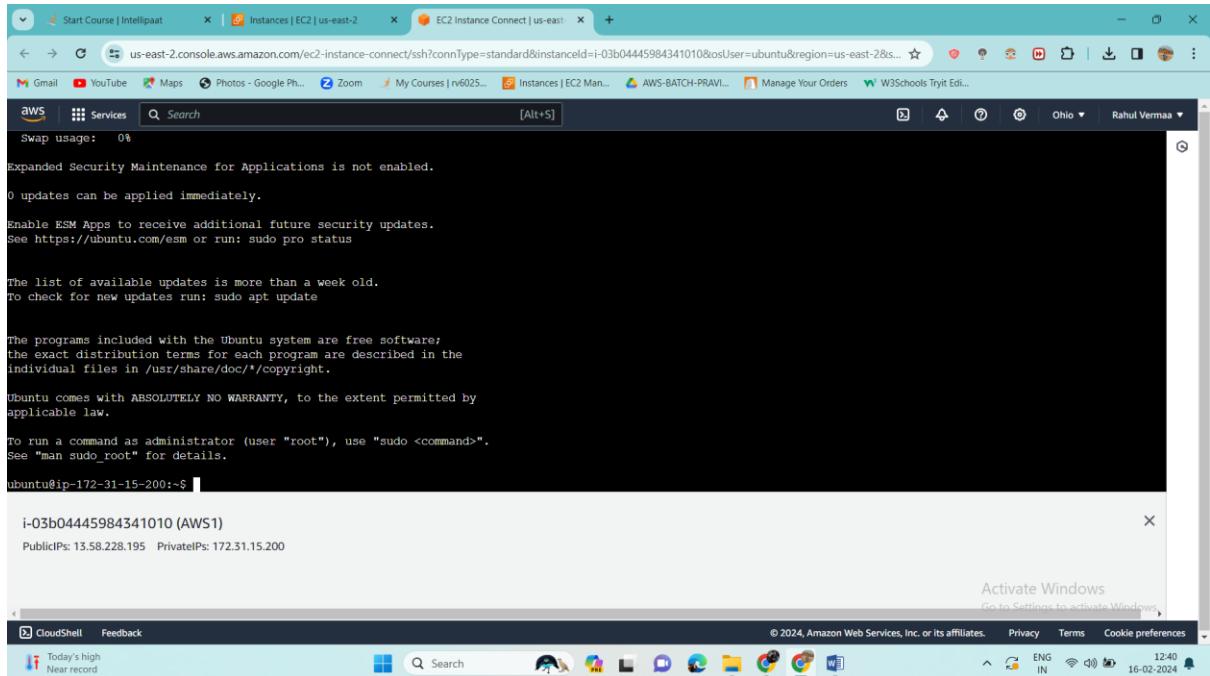


The screenshot shows the AWS EC2 Instances page. The left sidebar is expanded, showing categories like EC2 Dashboard, Instances, Images, and Elastic Block Store. The main content area is titled 'Instances Info' and displays a message: 'No instances' and 'You do not have any instances in this region'. A 'Launch instances' button is visible. Below the main content is a 'Select an instance' dropdown menu. The bottom of the screen shows the Windows taskbar with various pinned icons and the date/time as 16-02-2024.



The screenshot shows the AWS EC2 Instances page after an instance has been created. The left sidebar is expanded. The main content area is titled 'Instances (1) Info' and lists a single instance: 'AWS1' (Instance ID: i-03b04445984341010, State: Running, Type: t2.micro, Status: Initializing). The 'Select an instance' dropdown menu is still present. The bottom of the screen shows the Windows taskbar with various pinned icons and the date/time as 16-02-2024.

Now login to your ec2 instance



>First Update your system using the command-

**sudo apt-get update**

>Then use this command in Putty to install Apache2-

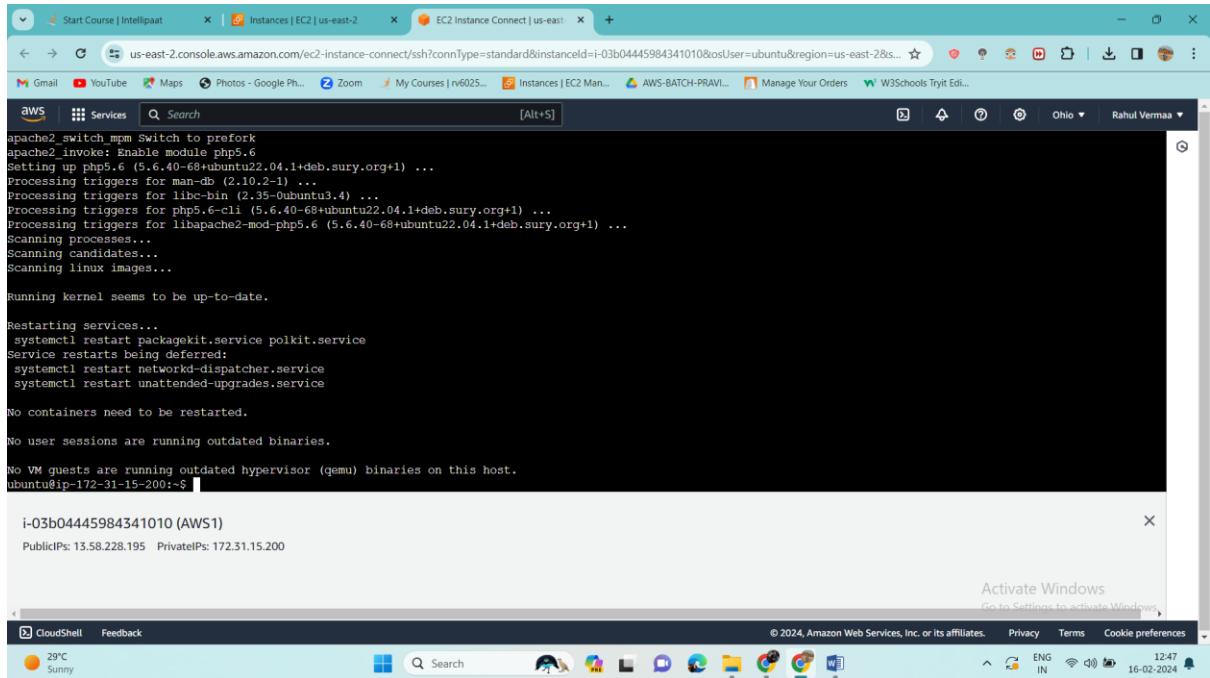
**sudo apt-get install apache2 -y**

> Then install php-mysql using the following command-

**sudo add-apt-repository -y ppa:ondrej/php**

**sudo apt install php5.6 mysql-client php5.6-mysql -y**

## Now everything is updated in your system



```
apache2: switch mpm Switch to prefork
apache2: invoke-rc.d: Enable module php5.6
Setting up php5.6 (5.6.40-68+ubuntu22.04.1+deb.sury.org+1) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.4) ...
Processing triggers for php5.6-cli (5.6.40-68+ubuntu22.04.1+deb.sury.org+1) ...
Processing triggers for libapache2-mod-php5.6 (5.6.40-68+ubuntu22.04.1+deb.sury.org+1) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.

Restarting services...
systemctl restart packagekit.service polkit.service
Service restarts being deferred:
systemctl restart network-dispatcher.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

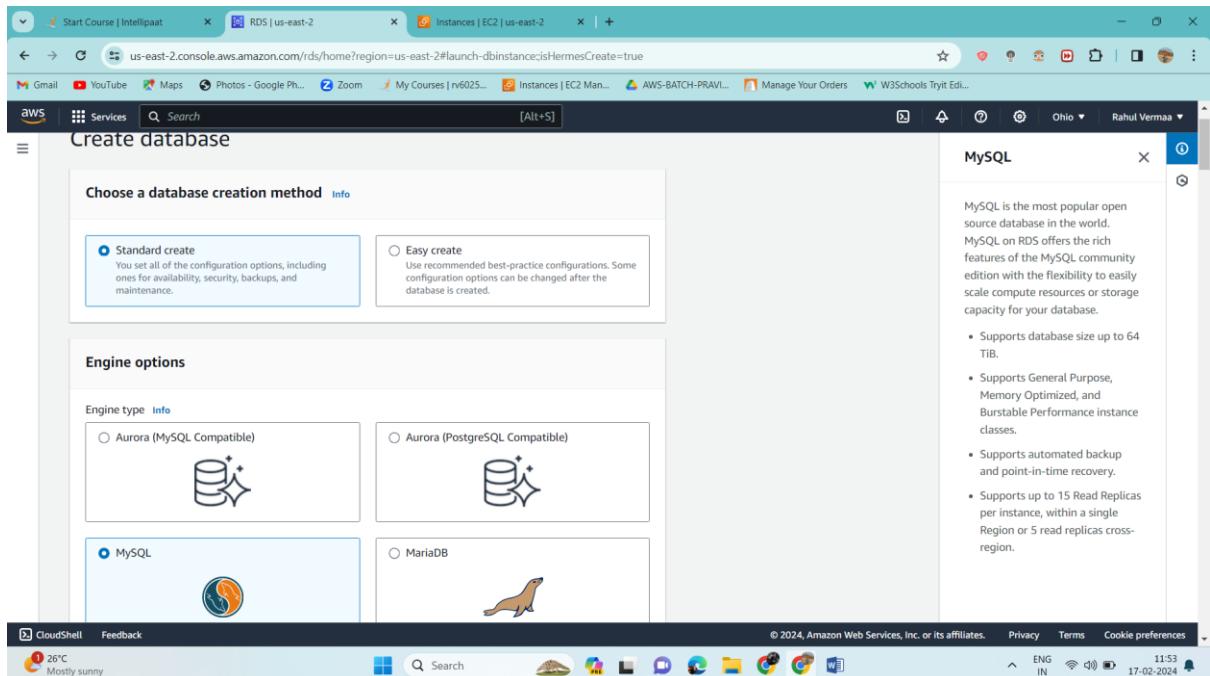
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.

ubuntu@ip-172-31-15-200:~$
```

i-03b04445984341010 (AWS1)  
PublicIPs: 13.58.228.195 PrivateIPs: 17.31.15.200

## Now create RDS



Choose a database creation method [Info](#)

Standard create  
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

Easy create  
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type [Info](#)

Aurora (MySQL Compatible) 

Aurora (PostgreSQL Compatible) 

MySQL 

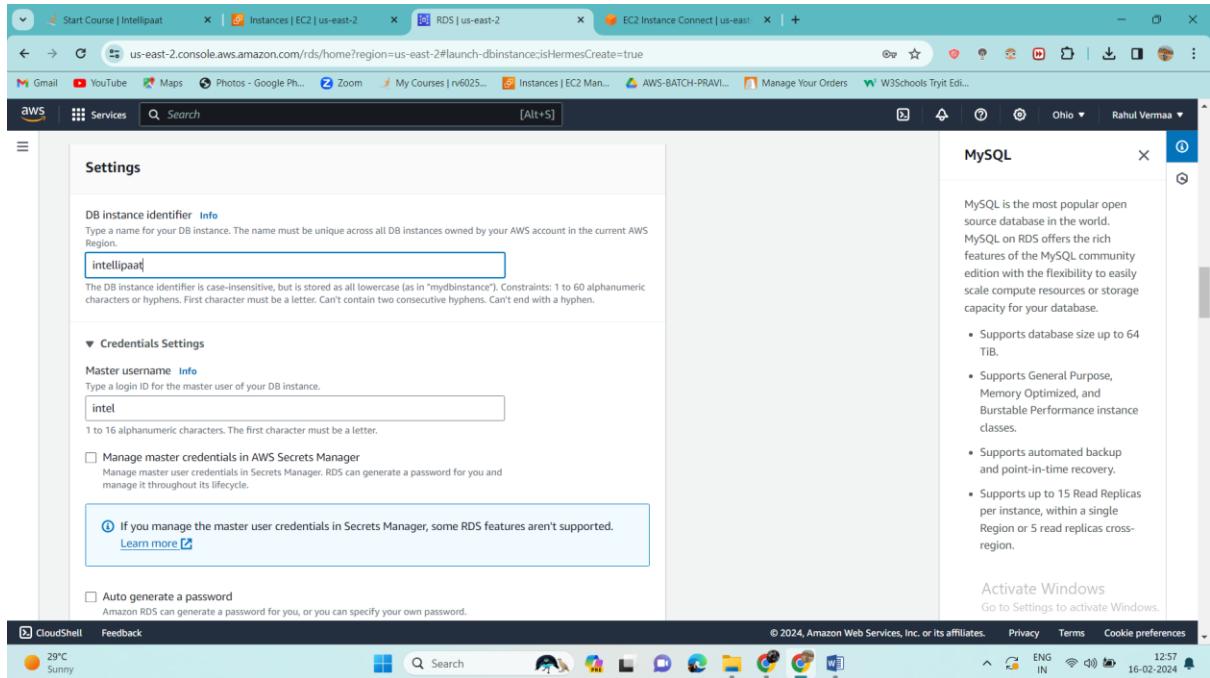
MariaDB 

**MySQL**

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

## Set password as- intel123



The screenshot shows the AWS RDS MySQL settings page. The 'DB instance identifier' is set to 'intellipaat'. The 'Master username' is set to 'intel'. The 'Auto generate a password' checkbox is unchecked. A note states: 'If you manage the master user credentials in Secrets Manager, some RDS features aren't supported. Learn more'.

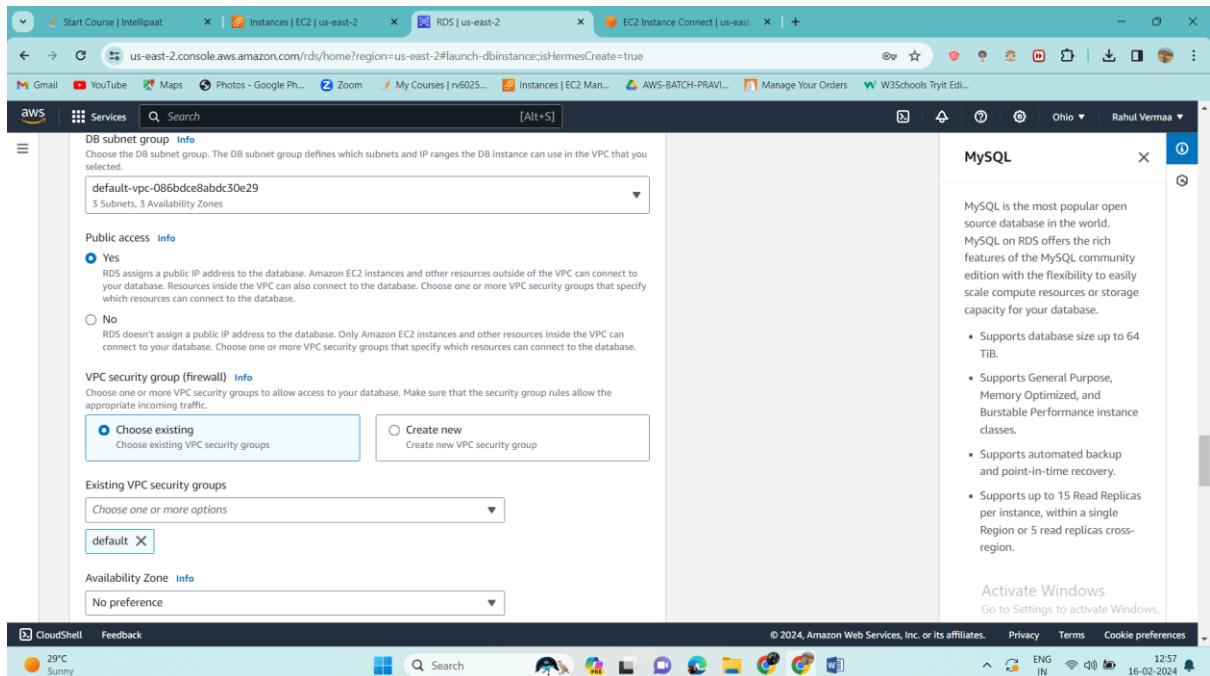
**MySQL**

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

Activate Windows  
Go to Settings to activate Windows.

## Default VPC



The screenshot shows the AWS RDS MySQL settings page. The 'DB subnet group' is set to 'default-vpc-086bdce8abd30e29'. The 'Public access' setting is 'Yes'. The 'VPC security group (firewall)' setting is 'Choose existing'. The 'Existing VPC security groups' dropdown shows 'default'. The 'Availability Zone' setting is 'No preference'.

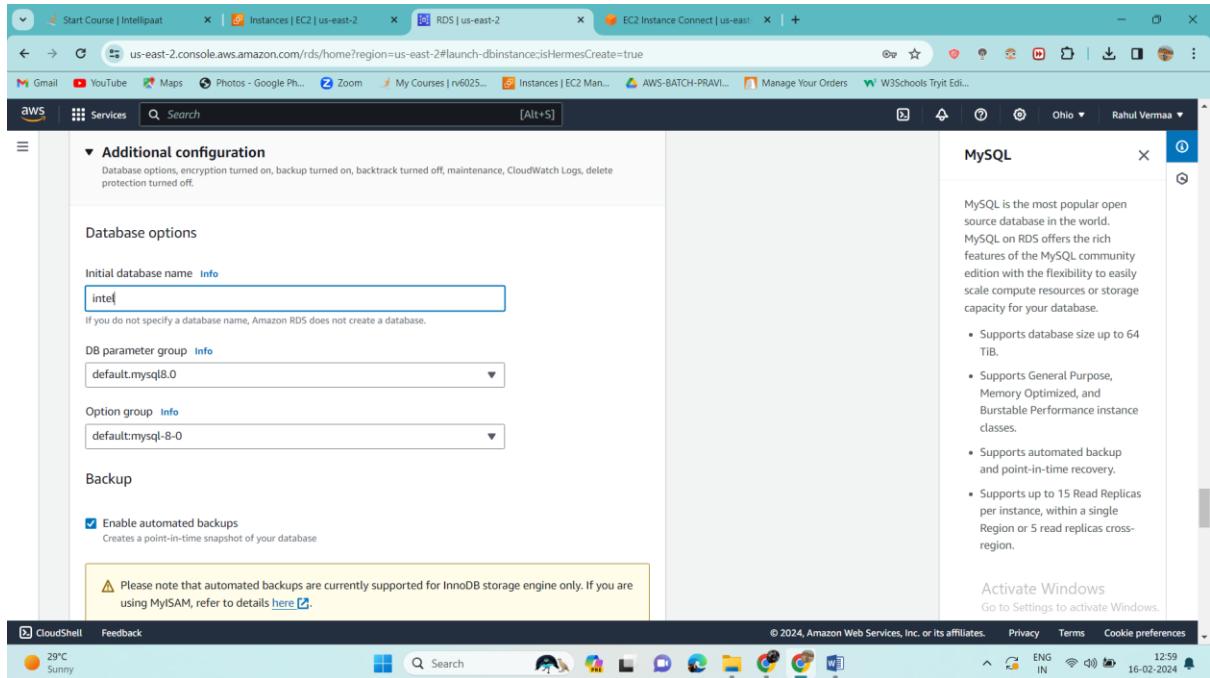
**MySQL**

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

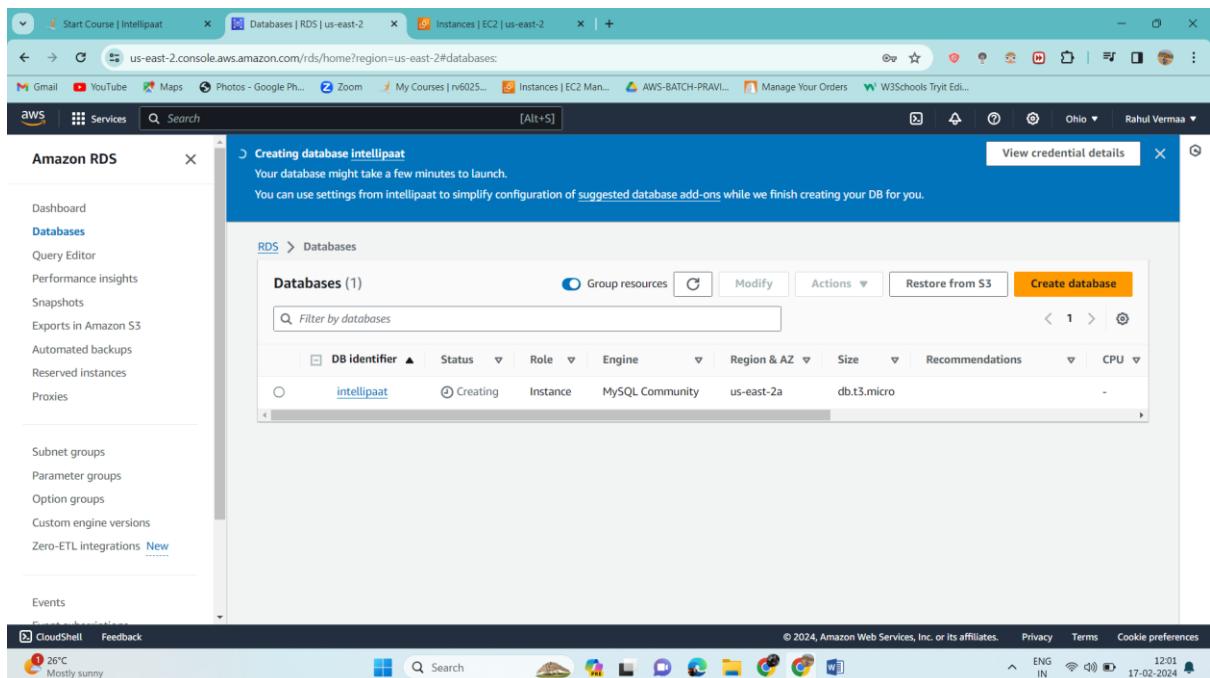
Activate Windows  
Go to Settings to activate Windows.

## Database name- intel



The screenshot shows the AWS RDS console with the 'Create database' wizard. The 'Initial database name' field is set to 'intel'. The 'DB parameter group' is 'default.mysql8.0' and the 'Option group' is 'default:mysql-8-0'. Under 'Backup', the 'Enable automated backups' checkbox is checked. A note states: '⚠ Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to details [here](#).' On the right, a 'MySQL' information panel is open, detailing MySQL as the most popular open-source database and listing its features. The AWS navigation bar at the top includes 'CloudShell', 'Feedback', and the user 'Rahul Vermaa'.

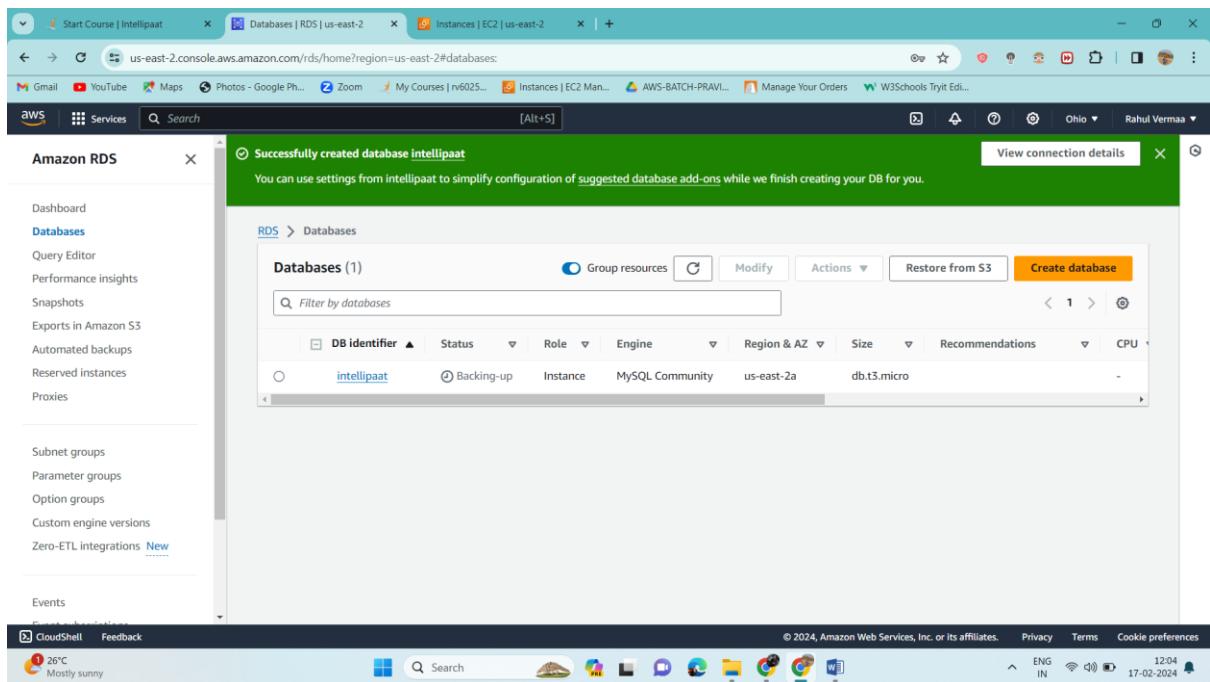
## RDS is created now



The screenshot shows the AWS RDS console with the 'Creating database intellipaat' status message. The database is currently 'Creating'. The 'Databases' table shows the following data:

DB identifier	Status	Role	Engine	Region & AZ	Size	Recommendations	CPU
intellipaat	Creating	Instance	MySQL Community	us-east-2a	db.t3.micro	-	-

The left sidebar shows the 'Amazon RDS' navigation menu with options like 'Dashboard', 'Databases', 'Performance insights', and 'Exports in Amazon S3'. The AWS navigation bar at the top includes 'CloudShell', 'Feedback', and the user 'Rahul Vermaa'.



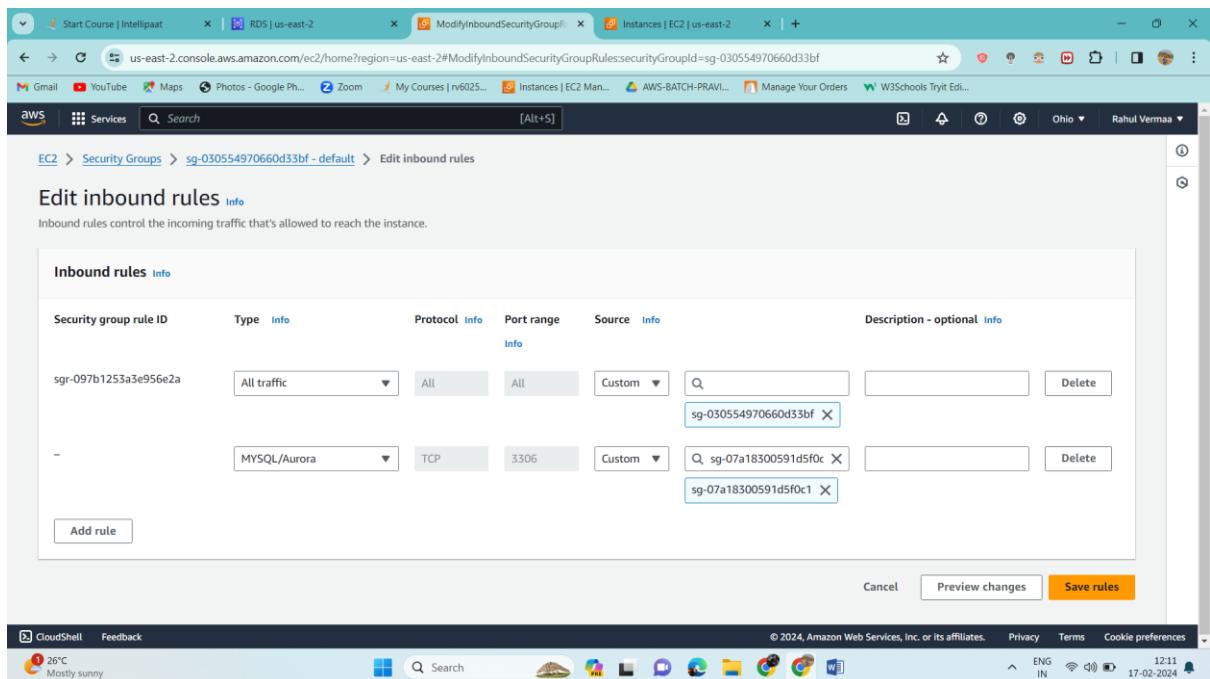
The screenshot shows the AWS RDS console with the following details:

- Header:** Start Course | Intellipaat, Databases | RDS | us-east-2, Instances | EC2 | us-east-2
- Breadcrumbs:** us-east-2.console.aws.amazon.com/rds/home?region=us-east-2#databases
- Left Sidebar:** Amazon RDS, Databases (selected), Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations (New), Events.
- Main Content:** Databases (1) table with one row:

DB identifier	Status	Role	Engine	Region & AZ	Size	Recommendations	CPU
intellipaat	Backing-up	Instance	MySQL Community	us-east-2a	db.t3.micro		
- Bottom:** CloudShell, Feedback, Weather (26°C, Mostly sunny), Search bar, AWS navigation icons, ENG IN, 12:04, 17-02-2024.

Now we have to configure RDS security group settings

Then paste the EC2 Security ID in Source > Custom > Security Group by keeping the Type as MySQL/Aurora



The screenshot shows the AWS EC2 ModifyInboundSecurityGroup interface with the following details:

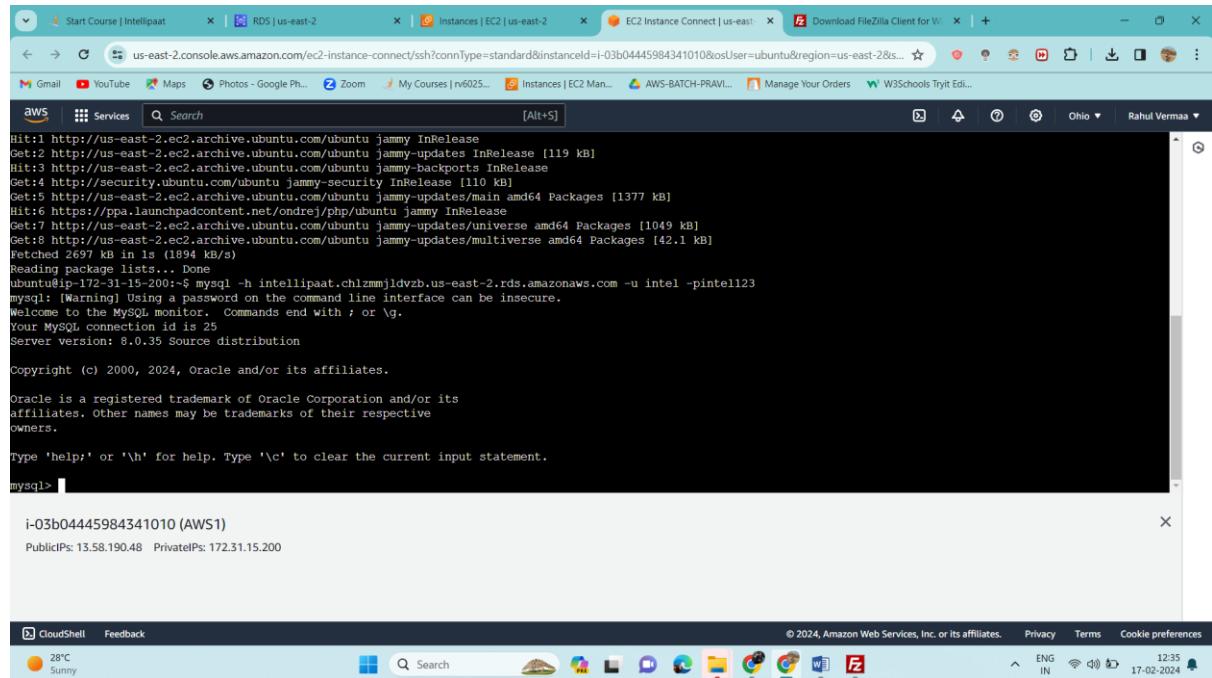
- Header:** Start Course | Intellipaat, RDS | us-east-2, ModifyInboundSecurityGroup, Instances | EC2 | us-east-2
- Breadcrumbs:** us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#ModifyInboundSecurityGroupRules:securityGroupId=sg-030554970660d33bf
- Section:** Edit inbound rules
- Table:** Inbound rules (Info)

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-097b1253a3e956e2a	All traffic	All	All	Custom	<input type="text"/> sg-030554970660d33bf <input type="button" value="Delete"/>
-	MySQL/Aurora	TCP	3306	Custom	<input type="text"/> sg-07a18300591d5f0c <input type="button" value="Delete"/>
- Buttons:** Add rule, Cancel, Preview changes, Save rules
- Bottom:** CloudShell, Feedback, Weather (26°C, Mostly sunny), Search bar, AWS navigation icons, ENG IN, 12:11, 17-02-2024.

Now we will connect to mysql using the below command

mysql -h (End point of RDS) -u (username- intel) -p(password without space- intel123)

-pintel123



```
hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
hit:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
get:5 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1377 kB]
hit:6 https://ppa.launchpadcontent.net/ondrej/php/ubuntu jammy InRelease
get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1049 kB]
Get:8 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [42.1 kB]
Fetched 2697 kB in 1s (1894 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-19-200:~$ mysql -h intellipaat.ch1zmm1dvzb.us-east-2.rds.amazonaws.com -u intel -pintel123
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 25
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

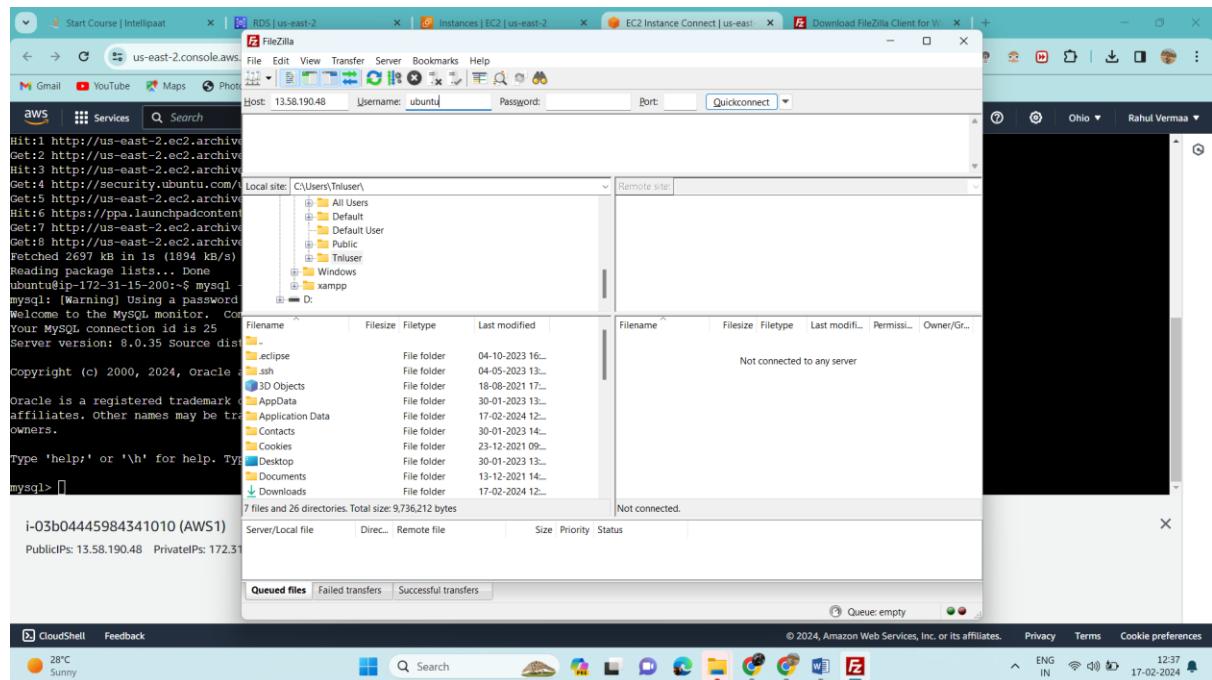
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

i-03b0445984341010 (AWS1)  
PublicIPs: 13.58.190.48 PrivateIPs: 172.31.15.200

Now we'll use filezilla Host- End point of ec2 and username- Ubuntu and click on connect



```
hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
hit:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
get:5 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1377 kB]
hit:6 https://ppa.launchpadcontent.net/ondrej/php/ubuntu jammy InRelease
get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1049 kB]
Get:8 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [42.1 kB]
Fetched 2697 kB in 1s (1894 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-19-200:~$ mysql -h intellipaat.ch1zmm1dvzb.us-east-2.rds.amazonaws.com -u intel -pintel123
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 25
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

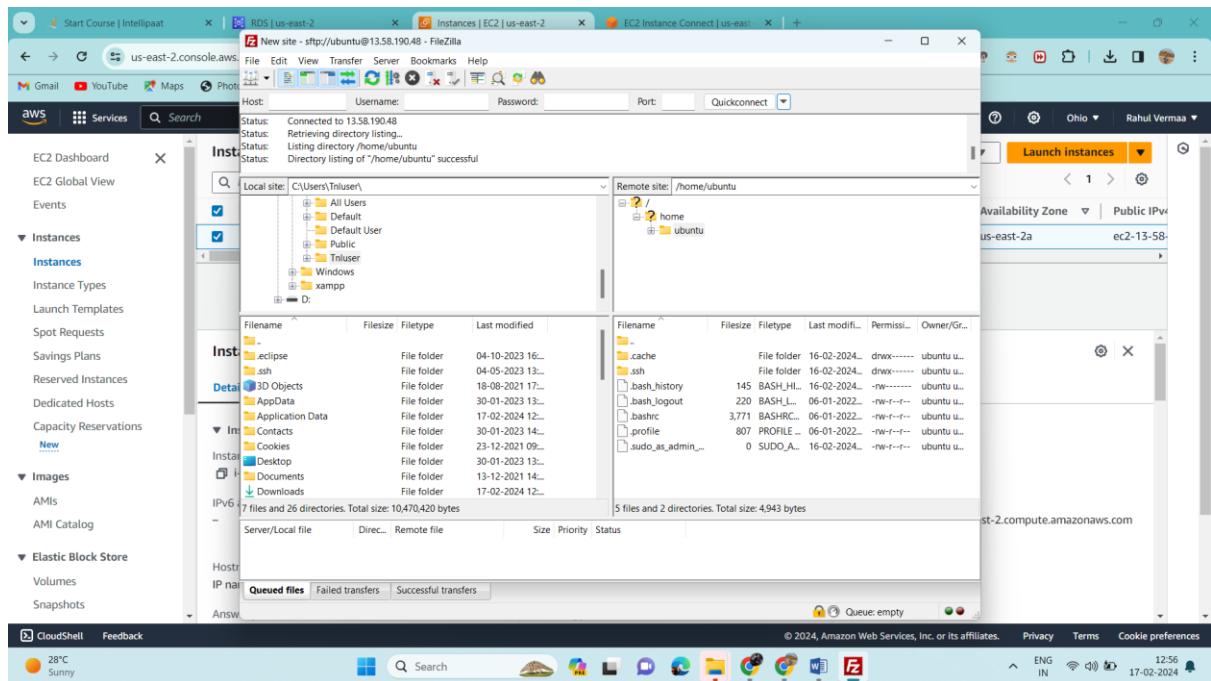
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

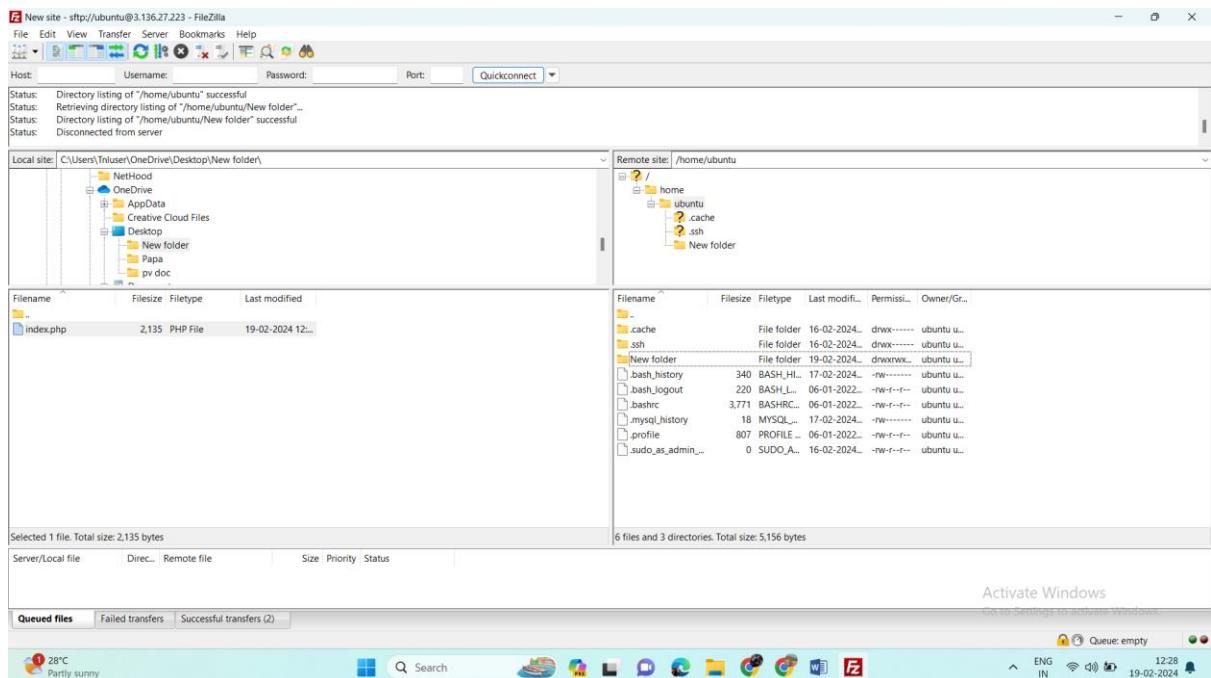
mysql> 
```

i-03b0445984341010 (AWS1)  
PublicIPs: 13.58.190.48 PrivateIPs: 172.31.15.200

## We are connect our ec2 with filezilla



Now we have created one 'New folder' and pasted our website in it

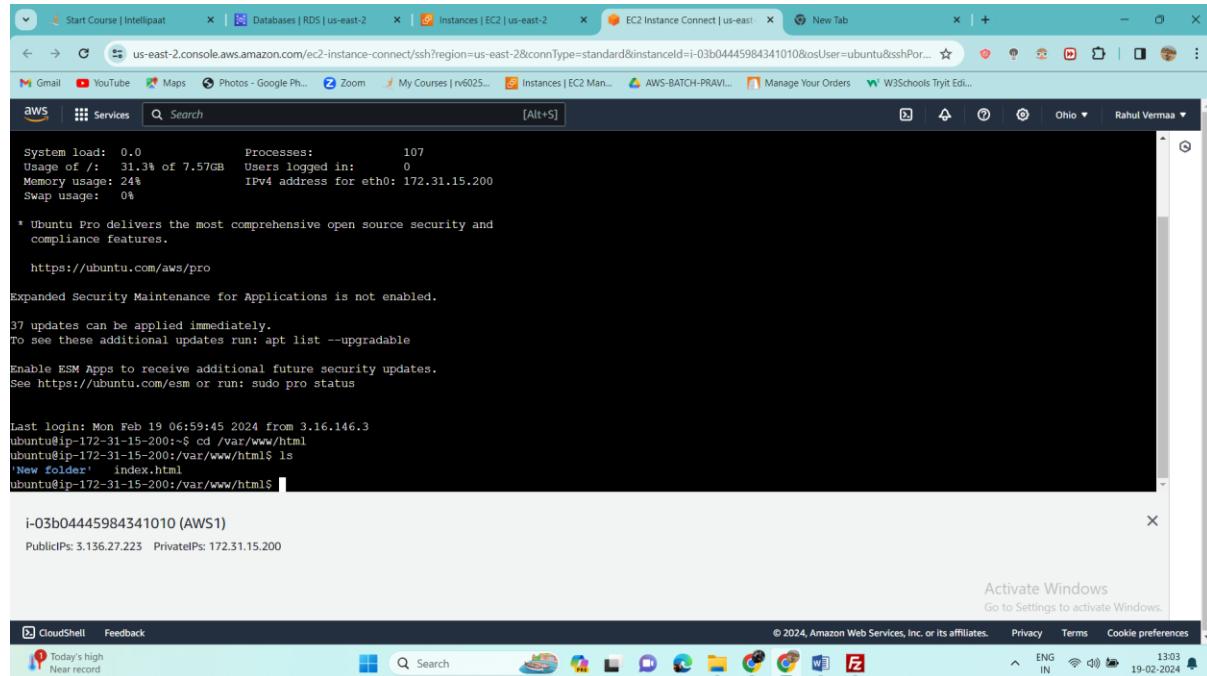


Now by using some commands will check our file

```
sudo cp -r New\ folder/ /var/www/html
```

```
cd /var/www/html
```

```
ls
```



```
system load: 0.0      Processes: 107
Usage of /: 31.3% of 7.57GB  Users logged in: 0
Memory usage: 24%      IPv4 address for eth0: 172.31.15.200
Swap usage: 0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

37 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

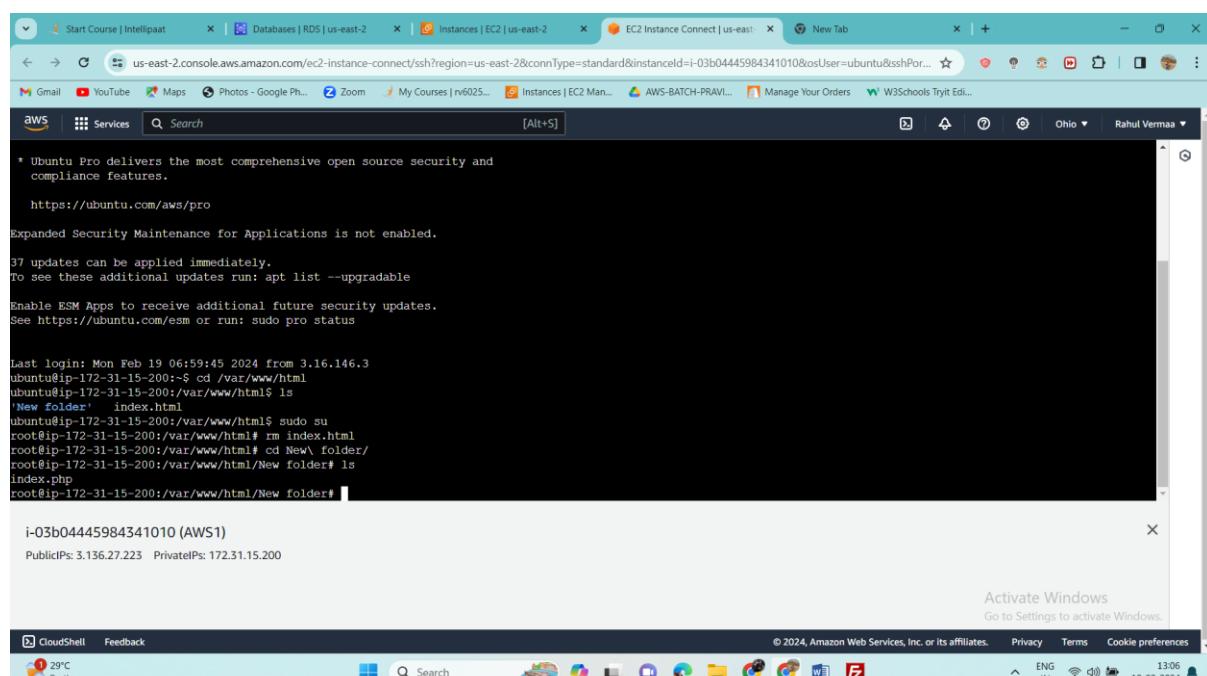
Last login: Mon Feb 19 06:59:45 2024 from 3.16.146.3
ubuntu@ip-172-31-15-200:~$ cd /var/www/html
ubuntu@ip-172-31-15-200:/var/www/html$ ls
'New folder'  index.html
ubuntu@ip-172-31-15-200:/var/www/html$ 

i-03b04445984341010 (AWS1)
PublicIPs: 3.136.27.223  PrivateIPs: 172.31.15.200

Activate Windows
Go to Settings to activate Windows.

CloudShell  Feedback  © 2024, Amazon Web Services, Inc. or its affiliates.  Privacy  Terms  Cookie preferences
Today's high
Near record  Search  ENG IN  13:06  19-02-2024
```

Now we will remove index.html file and will add index.php



```
* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

37 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

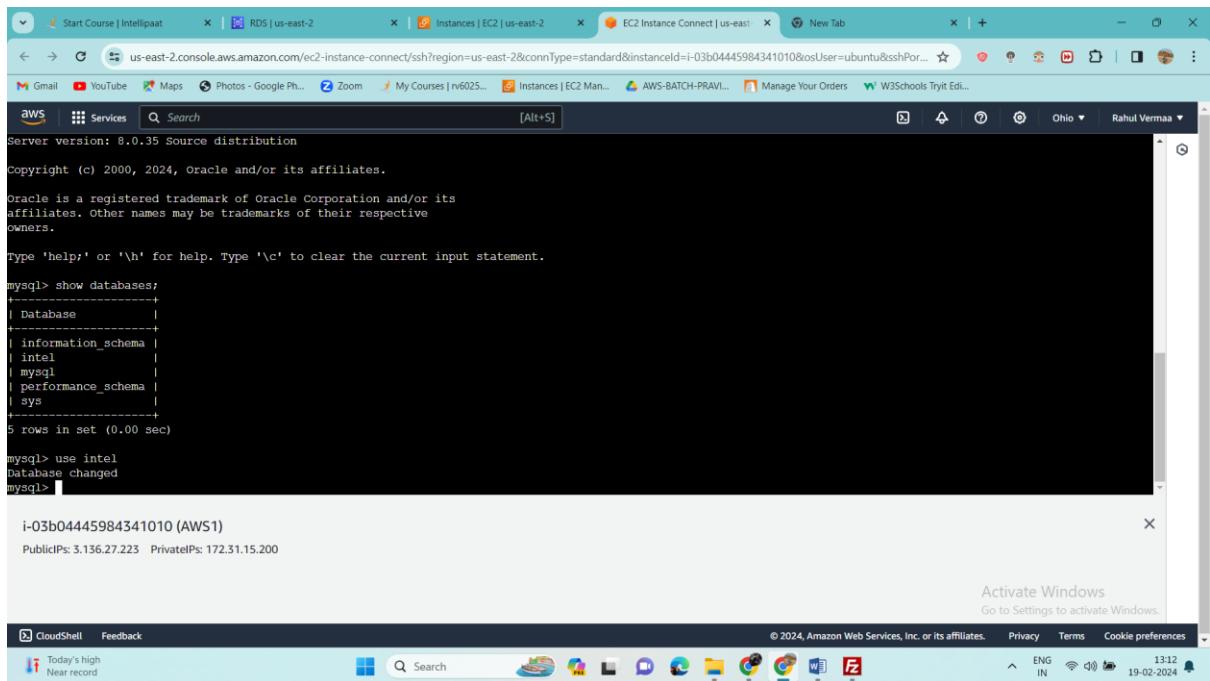
Last login: Mon Feb 19 06:59:45 2024 from 3.16.146.3
ubuntu@ip-172-31-15-200:~$ cd /var/www/html
ubuntu@ip-172-31-15-200:/var/www/html$ ls
'New folder'  index.html
ubuntu@ip-172-31-15-200:/var/www/html$ sudo su
root@ip-172-31-15-200:/var/www/html# rm index.html
root@ip-172-31-15-200:/var/www/html# cd New\ folder/
root@ip-172-31-15-200:/var/www/html/New folder# ls
index.php
root@ip-172-31-15-200:/var/www/html/New folder# 

i-03b04445984341010 (AWS1)
PublicIPs: 3.136.27.223  PrivateIPs: 172.31.15.200

Activate Windows
Go to Settings to activate Windows.

CloudShell  Feedback  © 2024, Amazon Web Services, Inc. or its affiliates.  Privacy  Terms  Cookie preferences
29°C  Partly sunny  Search  ENG IN  13:06  19-02-2024
```

## Now connect to mysql



```
Server version: 8.0.35 Source distribution
Copyright (c) 2000, 2024, 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> show databases;
+-----+
| Database |
+-----+
| information_schema |
| intel |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql> use intel
Database changed
mysql> 
```

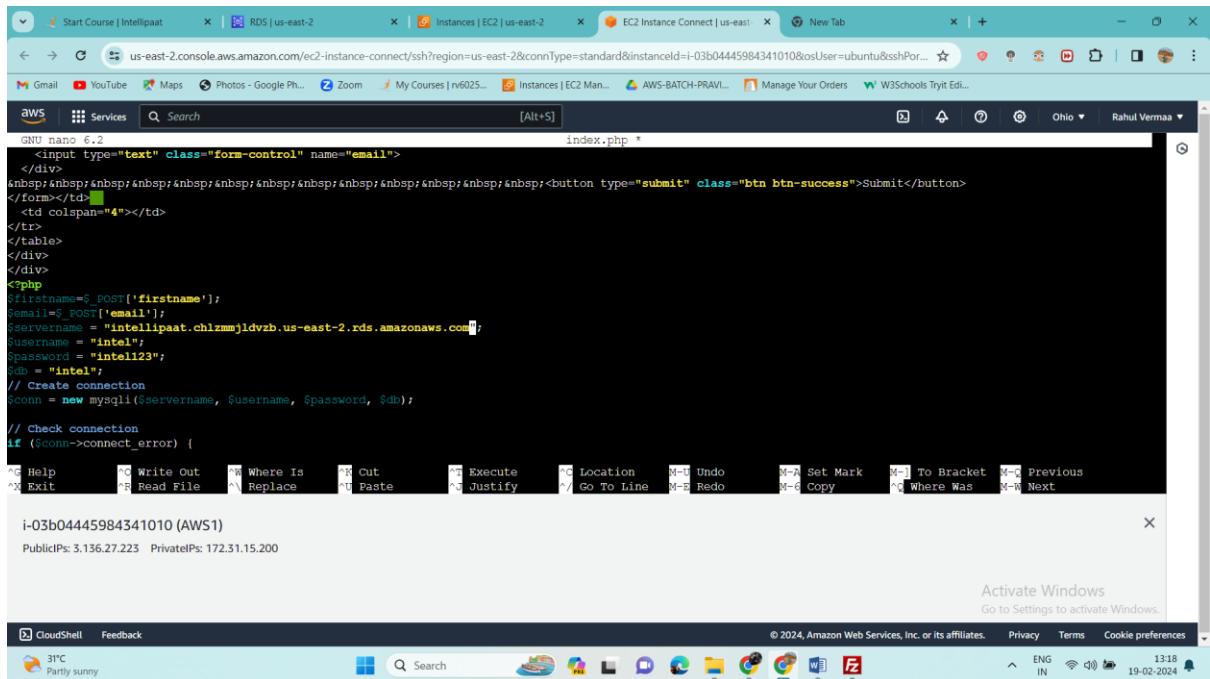
i-03b04445984341010 (AWS1)  
PublicIPs: 3.136.27.223 PrivateIPs: 172.31.15.200

Activate Windows  
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 13:12 19-02-2024

Now go to the path where website files are kept and run the index.php file by using sudo nano index.php

Now after this, GNU nano will pop up where you have to make changes in your code, you have to check if in your server name, the endpoint of your RDS is there along with username, password and db name



```
GNU nano 6.2 index.php *
<input type="text" class="form-control" name="email">
</div>
<form><td><button type="submit" class="btn btn-success">Submit</button>
</td><td colspan="4"></td>
</tr>
</table>
</div>
</div>
<?php
$firstname=$_POST['firstname'];
$email=$_POST['email'];
$servername = "intellipaat.chlzzmijldvzb.us-east-2.rds.amazonaws.com";
$username = "intel";
$password = "intel123";
$db = "intel";
// Create connection
$conn = new mysqli($servername, $username, $password, $db);

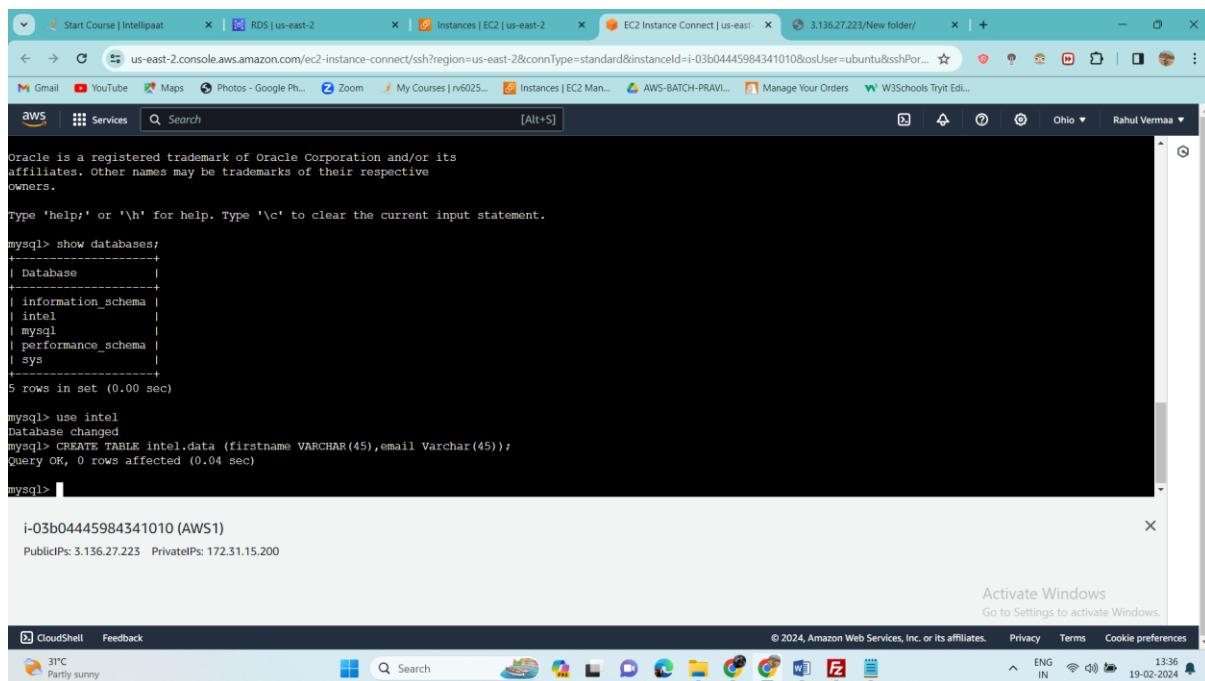
// Check connection
if ($conn->connect_error) {
```

i-03b04445984341010 (AWS1)  
PublicIPs: 3.136.27.223 PrivateIPs: 172.31.15.200

Activate Windows  
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 13:18 19-02-2024

## We have created one table in our intel databases



```
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> show databases;
+--------------------+
| Database          |
+--------------------+
| information_schema |
| intel              |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> use intel
Database changed
mysql> CREATE TABLE intel.data (firstname VARCHAR(45),email VARCHAR(45));
Query OK, 0 rows affected (0.04 sec)

mysql> |
```

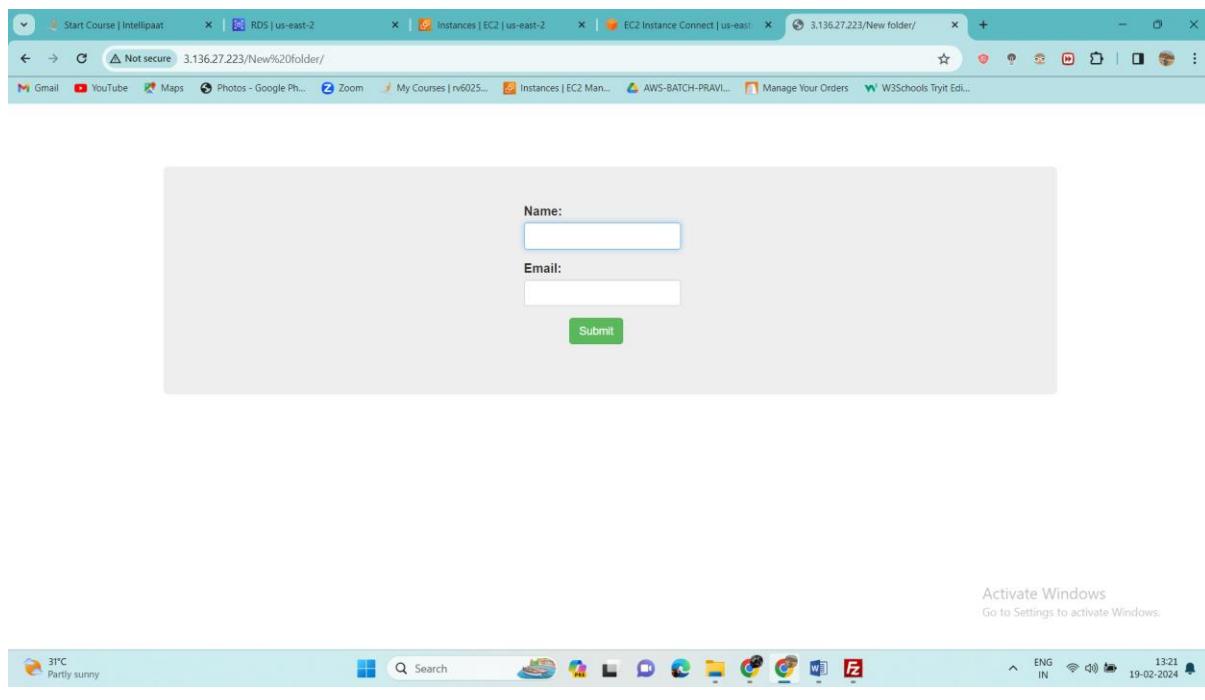
i-03b04445984341010 (AWS1)  
PublicIPs: 3.136.27.223 PrivateIPs: 172.31.15.200

Activate Windows  
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

31°C Partly sunny 13:36 19-02-2024

Now when you will try, and copy paste the Public IP of your EC2 Instance



Not secure 3.136.27.223/New%20folder/

Name:

Email:

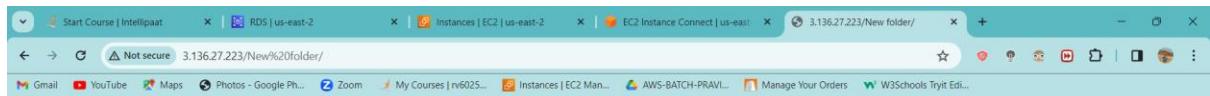
Submit

Activate Windows  
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

31°C Partly sunny 13:21 19-02-2024

Now let's add some data in it



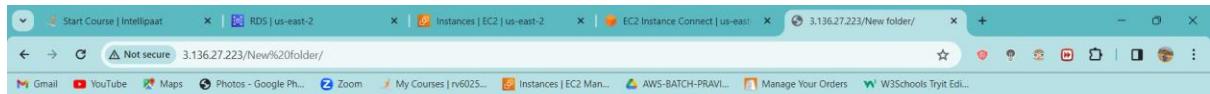
Name:

Email:

Activate Windows  
Go to Settings to activate Windows.



And our data is added successfully



Name:

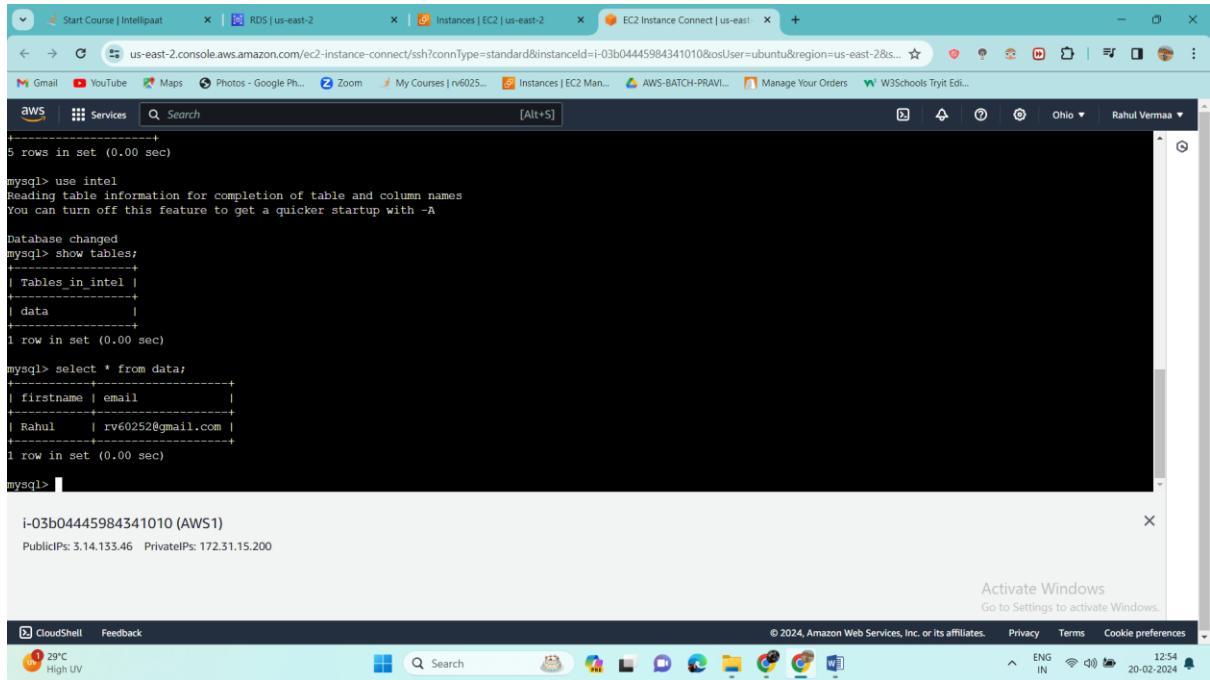
Email:

New record created successfully

Activate Windows  
Go to Settings to activate Windows.



Let's check it in our my sql



```
Start Course | Intellipaat | RDS | us-east-2 | Instances | EC2 | us-east-2 | EC2 Instance Connect | us-east-2 | + | us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-03b04445984341010&osUser=ubuntu&region=us-east-2&s... | Gmail | YouTube | Maps | Photos - Google Ph... | Zoom | My Courses | rv6025... | Instances | EC2 Man... | AWS-BATCH-PRAVI... | Manage Your Orders | W3Schools TryIt Ed... | AWS Services | Search | [Alt+S] | Ohio | Rahul Vermaa | + | 5 rows in set (0.00 sec)

mysql> use intel
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

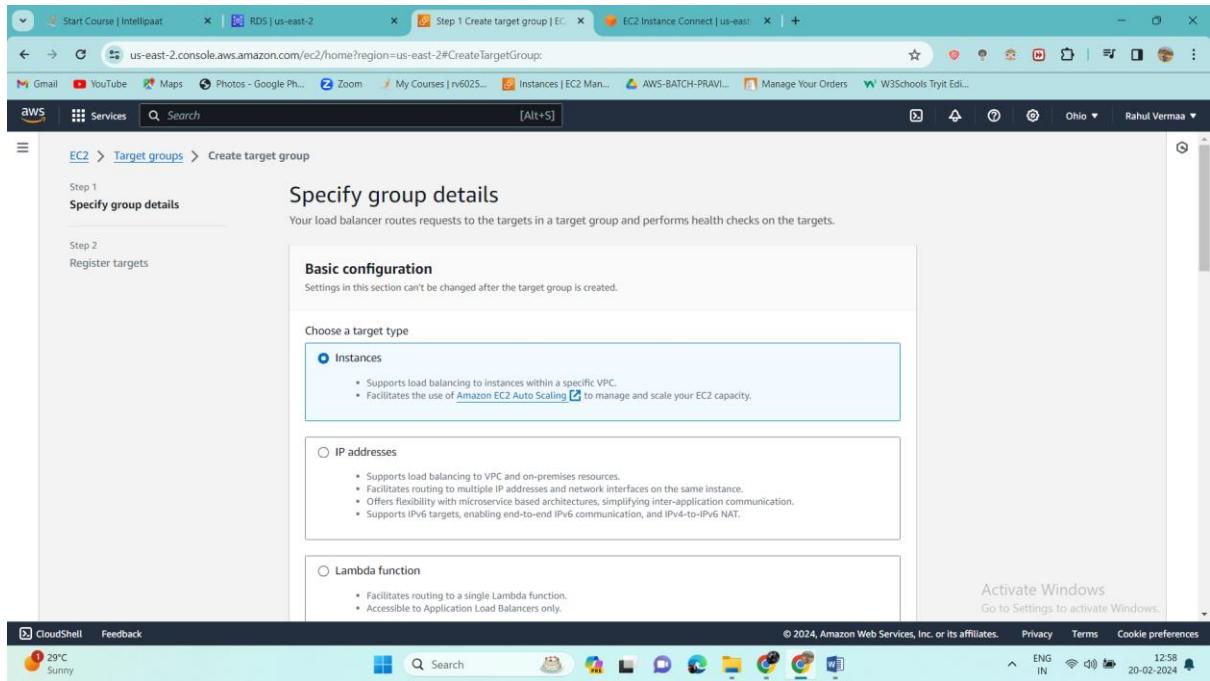
Database changed
mysql> show tables;
+-----+
| Tables in intel |
+-----+
| data |
+-----+
1 row in set (0.00 sec)

mysql> select * from data;
+-----+
| firstname | email |
+-----+
| Rahul | rv60252@gmail.com |
+-----+
1 row in set (0.00 sec)

mysql> | i-03b04445984341010 (AWS1)
PublicIPs: 3.14.153.46 PrivateIPs: 172.31.15.200 | x | Activate Windows | Go to Settings to activate Windows. | © 2024, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences | CloudShell | Feedback | 29°C | High UV | Search | ENG IN | 20-02-2024 | 12:54 |
```

The screenshot shows a CloudShell terminal window with the AWS Services bar at the top. The terminal is running a MySQL session on an EC2 instance named 'AWS1'. The user runs 'use intel', 'show tables;', and 'select \* from data;'. The output shows a single table 'data' with one row containing 'Rahul' and 'rv60252@gmail.com'. The AWS navigation bar at the bottom includes links for CloudShell, Feedback, and a weather widget showing 29°C and High UV.

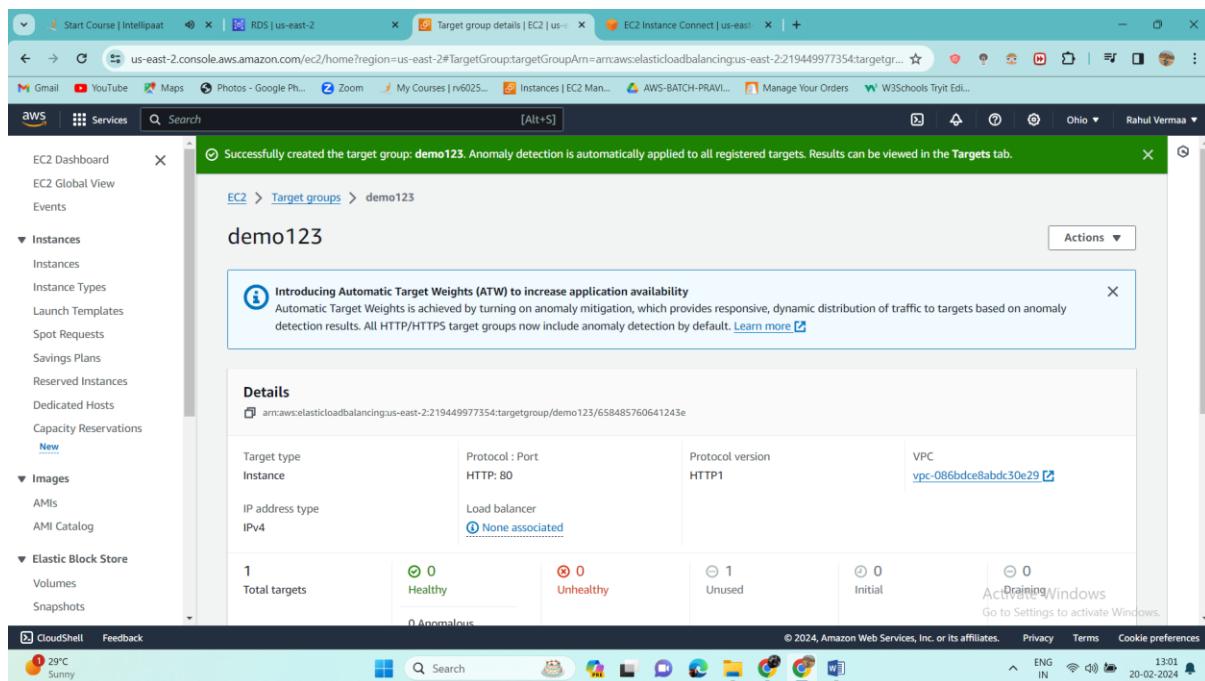
Now we will create load balancer, before that we will create Target group



```
Start Course | Intellipaat | RDS | us-east-2 | Step 1 Create target group | EC2 | EC2 Instance Connect | us-east-2 | + | us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#CreateTargetGroup; | Gmail | YouTube | Maps | Photos - Google Ph... | Zoom | My Courses | rv6025... | Instances | EC2 Man... | AWS-BATCH-PRAVI... | Manage Your Orders | W3Schools TryIt Ed... | AWS Services | Search | [Alt+S] | Ohio | Rahul Vermaa | + | Step 1 | Step 2 | Target groups | Create target group | EC2 > Target groups > Create target group | Step 1 | Specify group details | Step 2 | Register targets | Basic configuration | Choose a target type | Instances | IP addresses | Lambda function | Activate Windows | Go to Settings to activate Windows. | © 2024, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences | CloudShell | Feedback | 29°C | Sunny | Search | ENG IN | 20-02-2024 | 12:58 |
```

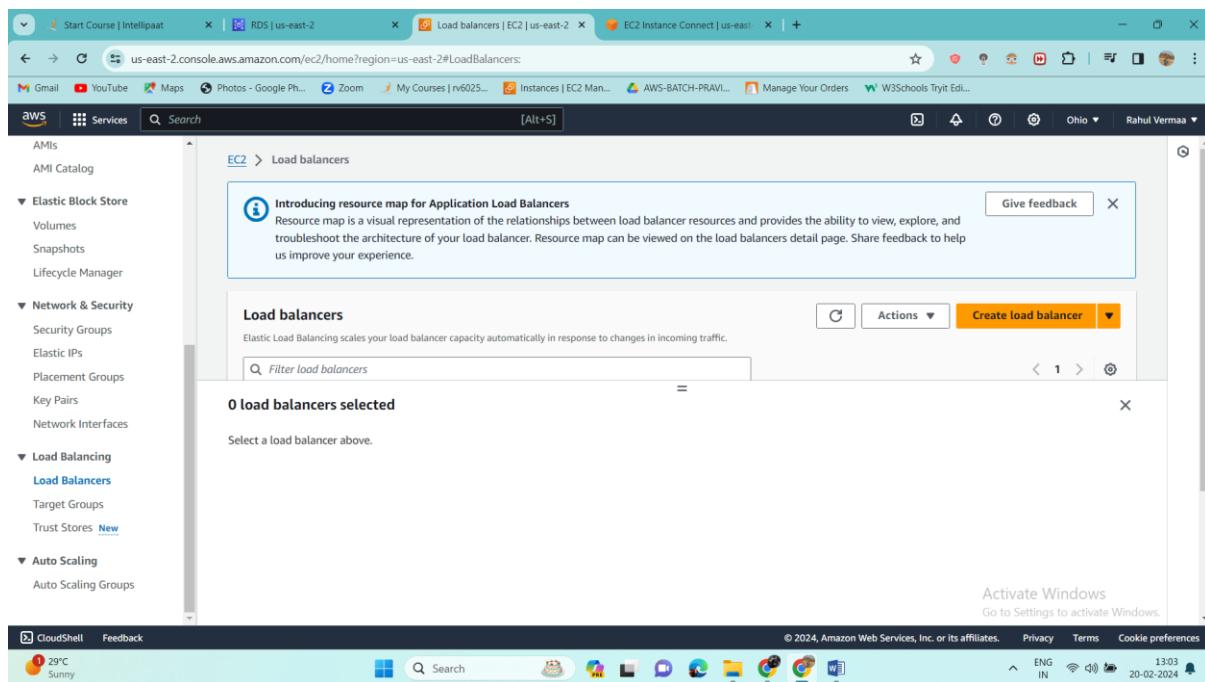
The screenshot shows the 'Create target group' wizard in the AWS EC2 console. The user is on Step 1: Specify group details. The 'Basic configuration' section is visible, showing the 'Choose a target type' dropdown with 'Instances' selected. Other options include 'IP addresses' and 'Lambda function'. The AWS navigation bar at the bottom includes links for CloudShell, Feedback, and a weather widget showing 29°C and Sunny.

## We have added our instance and our target group is created



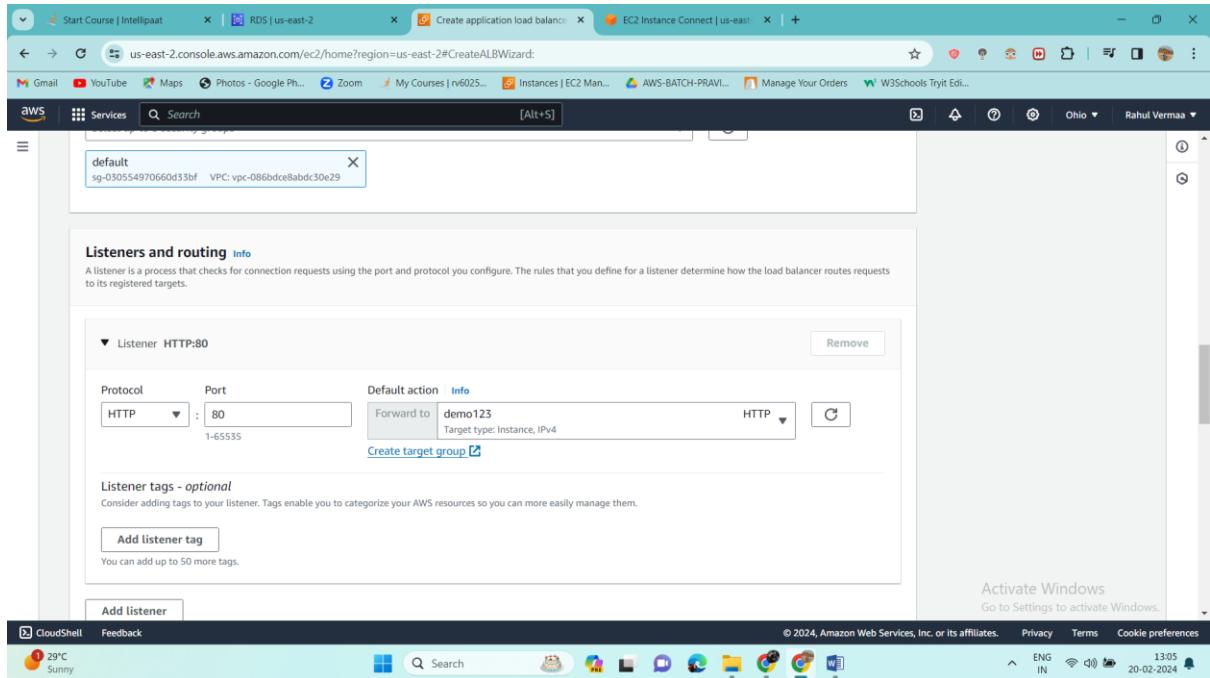
The screenshot shows the AWS Cloud Console with the URL [us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#TargetGroup:targetGroupArn=arn:aws:elasticloadbalancing:us-east-2:219449977354:targetgroup/demo123](https://us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#TargetGroup:targetGroupArn=arn:aws:elasticloadbalancing:us-east-2:219449977354:targetgroup/demo123). The page displays a success message: "Successfully created the target group: demo123. Anomaly detection is automatically applied to all registered targets. Results can be viewed in the Targets tab." The left sidebar shows the EC2 dashboard with various navigation options like Instances, Images, and Elastic Block Store. The main content area shows the "demo123" target group details, including its ARN, target type (Instance), protocol (HTTP: 80), and VPC (vpc-086bdce8abdc30e29). It also shows 1 total target, 0 healthy, 0 unhealthy, 1 unused, 0 initial, and 0 draining. A note at the bottom right says "Activate Windows" and "Go to Settings to activate Windows".

## Now we can create our load balancer



The screenshot shows the AWS Cloud Console with the URL [us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#LoadBalancers](https://us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#LoadBalancers). The left sidebar includes the Load Balancing section with "Load Balancers" selected. The main content area shows a message about the resource map for Application Load Balancers. Below it, the "Load balancers" section shows 0 load balancers selected. A "Create load balancer" button is prominently displayed. A note at the bottom right says "Activate Windows" and "Go to Settings to activate Windows".

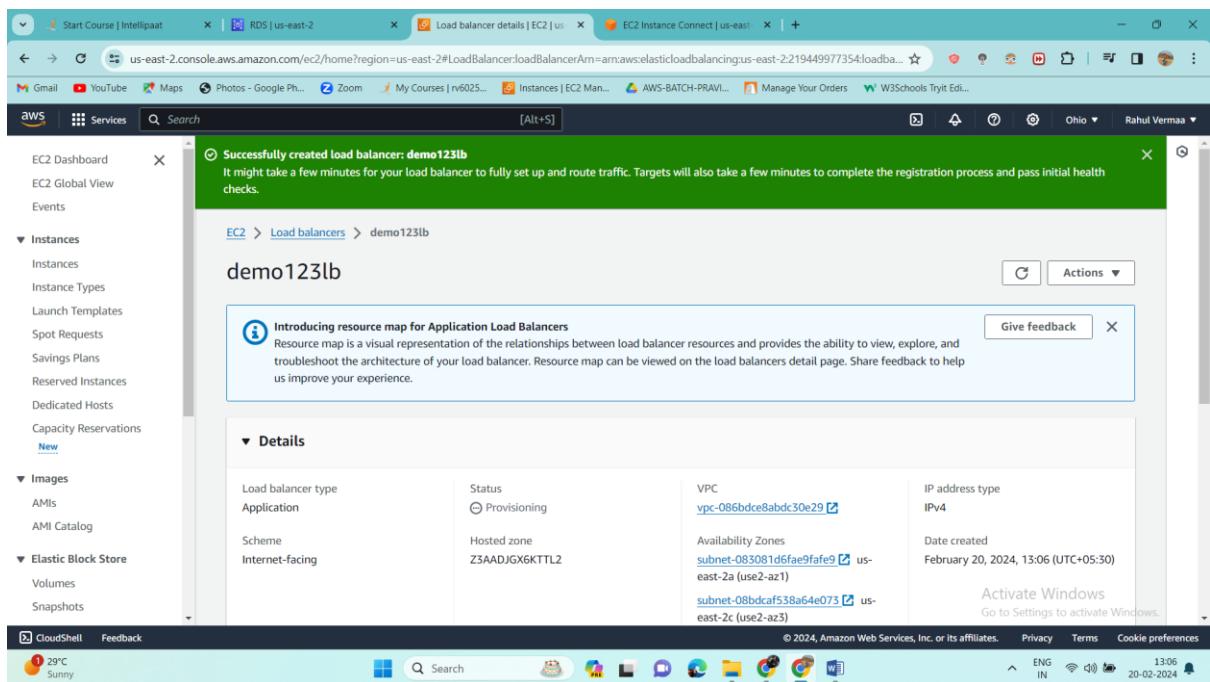
Now everything default we just have to select our target group



The screenshot shows the AWS CloudWatch Metrics console. A log stream named 'demo123' is selected. The log entries are timestamped and show the creation of a target group. The most recent entry is:

```
2024-02-20T13:05:53.123Z demo123 [INFO] Target group created: arn:aws:elasticloadbalancing:us-east-2:2219449977354:loadbalancer-targetgroup/dem...  
2024-02-20T13:05:53.123Z demo123 [INFO] Target group created: arn:aws:elasticloadbalancing:us-east-2:2219449977354:loadbalancer-targetgroup/dem...
```

And our load balancer is also created

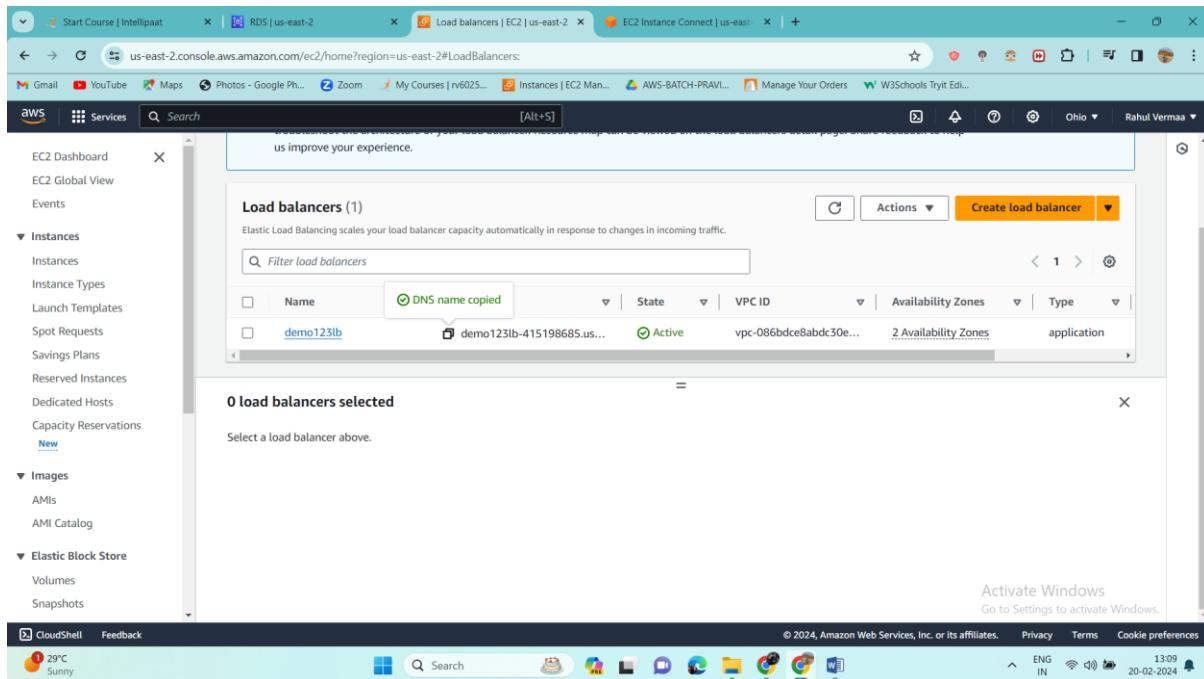


The screenshot shows the AWS CloudWatch Metrics console. A log stream named 'demo123' is selected. The log entries are timestamped and show the creation of a target group. The most recent entry is:

```
2024-02-20T13:06:06.123Z demo123 [INFO] Target group created: arn:aws:elasticloadbalancing:us-east-2:2219449977354:loadbalancer-targetgroup/dem...  
2024-02-20T13:06:06.123Z demo123 [INFO] Target group created: arn:aws:elasticloadbalancing:us-east-2:2219449977354:loadbalancer-targetgroup/dem...
```

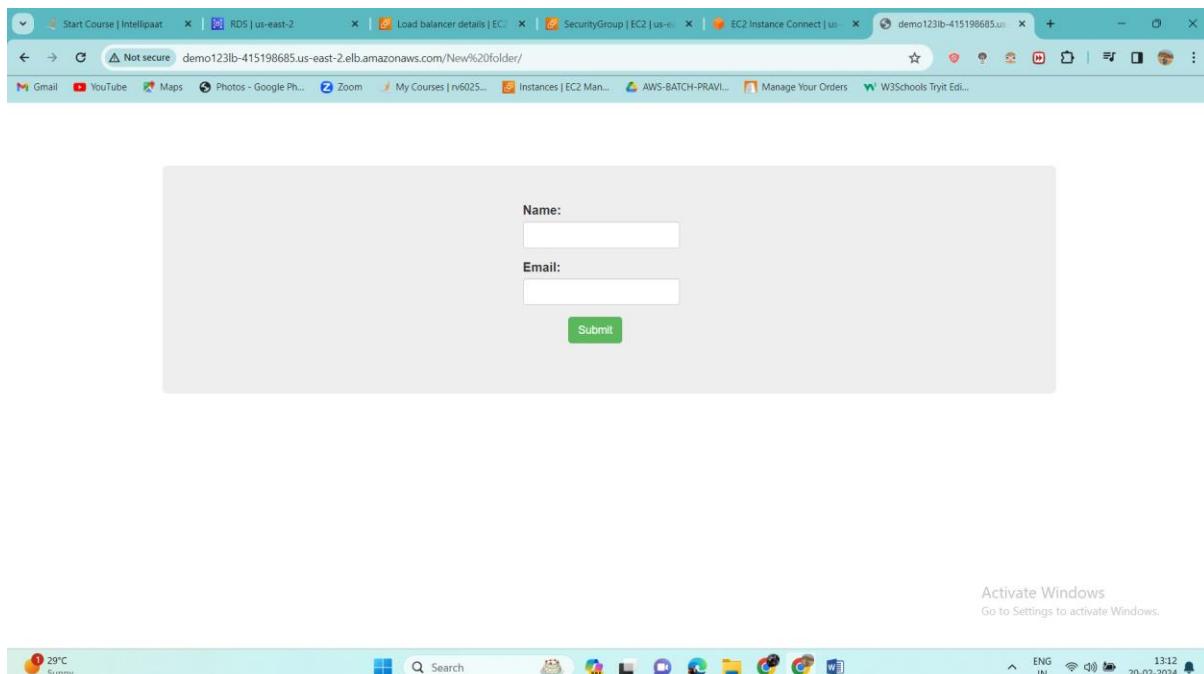
Now let's check if it's working fine let's copy it's DNS and paste it in browser

It should show our website



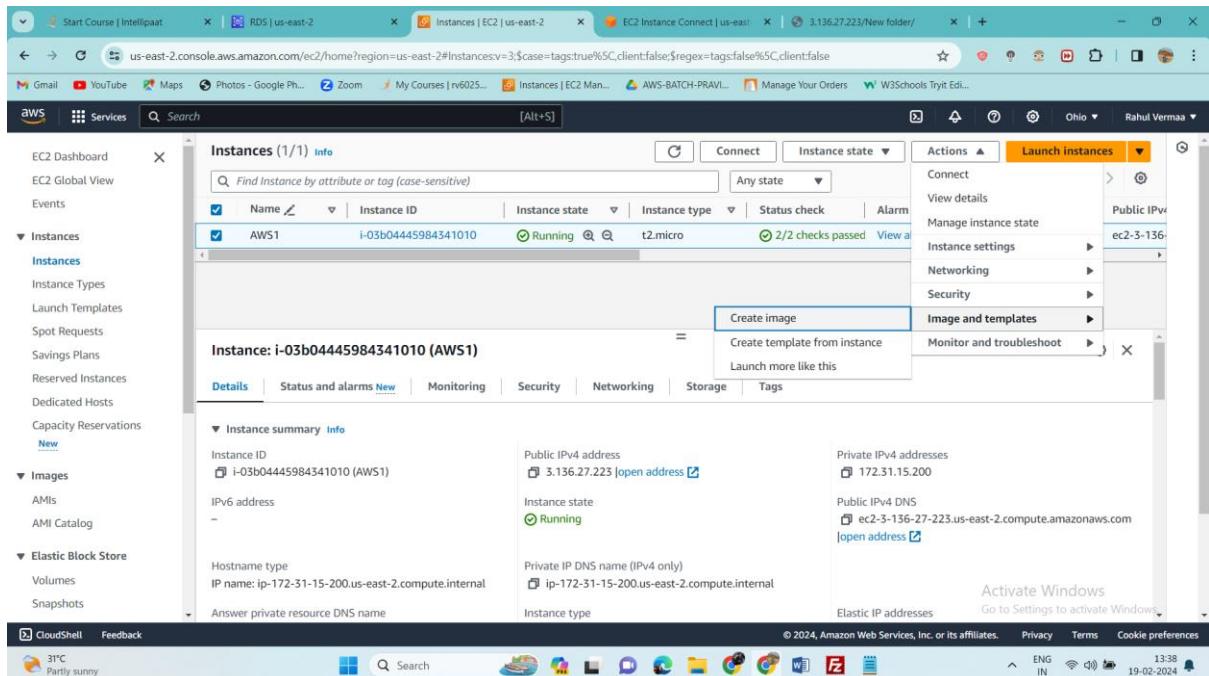
The screenshot shows the AWS CloudWatch Metrics Insights interface. A query is being run: `aws cloudwatch metrics insights Metrics`. The results table is empty, showing 0 rows. The interface includes a sidebar with navigation links like EC2 Dashboard, EC2 Global View, Instances, Images, and Elastic Block Store. The top navigation bar shows tabs for Start Course, RDS, Load balancers, EC2 Instance Connect, and EC2 home. The status bar at the bottom indicates the region is us-east-2 and the time is 20-02-2024.

And it's working properly



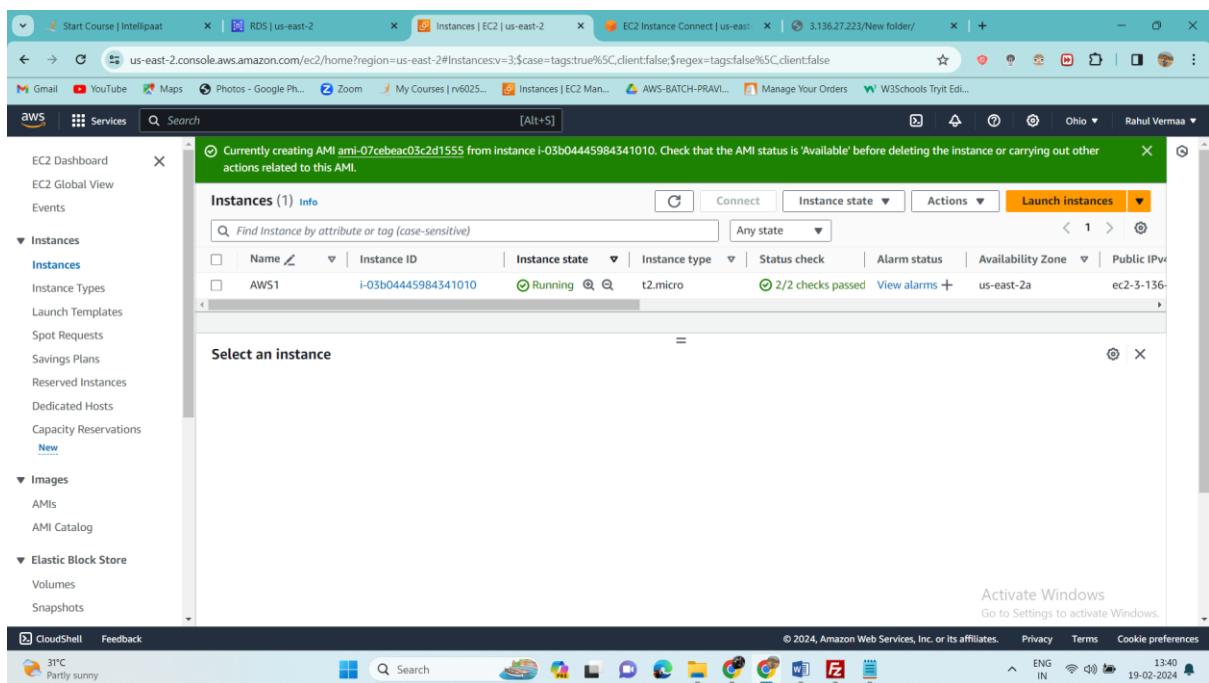
The screenshot shows a web browser displaying the output of a POST request to a CloudWatch Metrics Insights endpoint. The response body contains a JSON object with fields: `version`, `status`, `error`, `data`, and `logs`. The `data` field contains a single item with a timestamp of `2024-02-20T13:12:00Z` and a value of `1`. The browser's address bar shows the URL: `demo123lb-415198685.us-east-2.elb.amazonaws.com/New%20folder/`. The status bar at the bottom indicates the region is us-east-2 and the time is 20-02-2024.

Now, we'll do the autoscaling of our website by going to our EC2 Instance and then click on Actions and Create Image



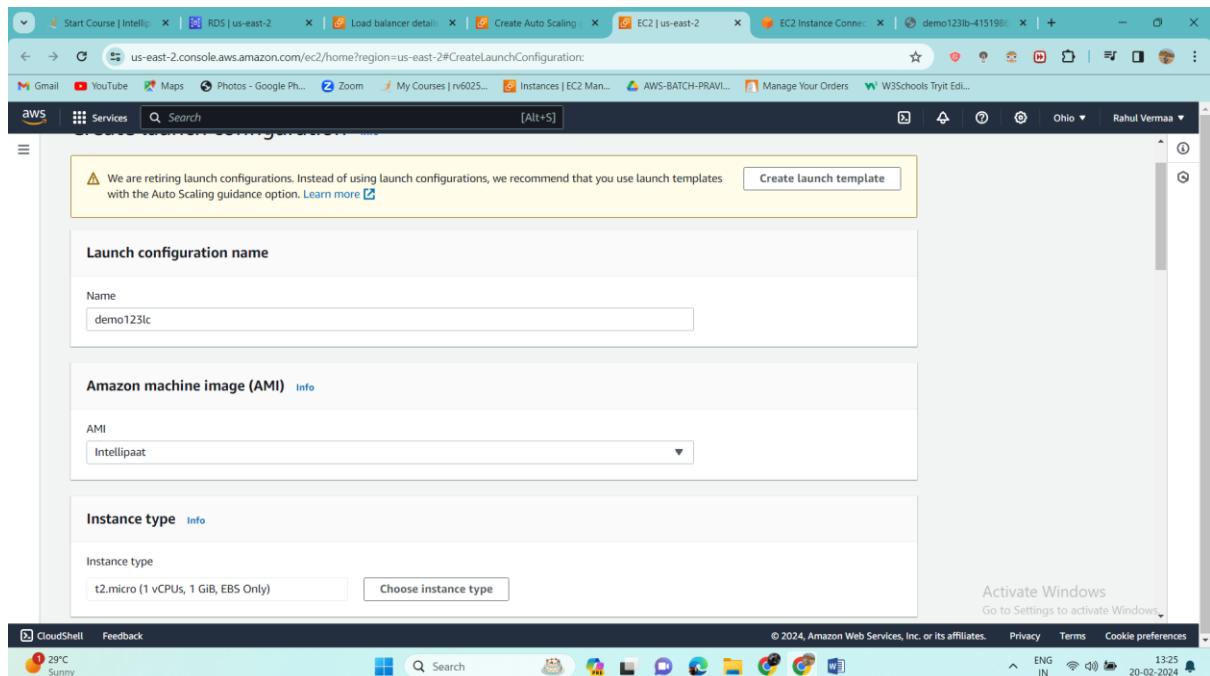
The screenshot shows the AWS EC2 Instances page. On the left, a sidebar lists various EC2-related options like Instances, Images, and Elastic Block Store. The main area shows a table of instances with one row selected for 'AWS1'. The 'Actions' menu is open, and the 'Create image' option is highlighted with a blue border. Below the table, there are tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The 'Details' tab is selected. On the right, there are sections for Instance summary, IP addresses, and Elastic IP addresses. The status bar at the bottom shows the date as 19-02-2024 and the time as 13:38.

Our image is created

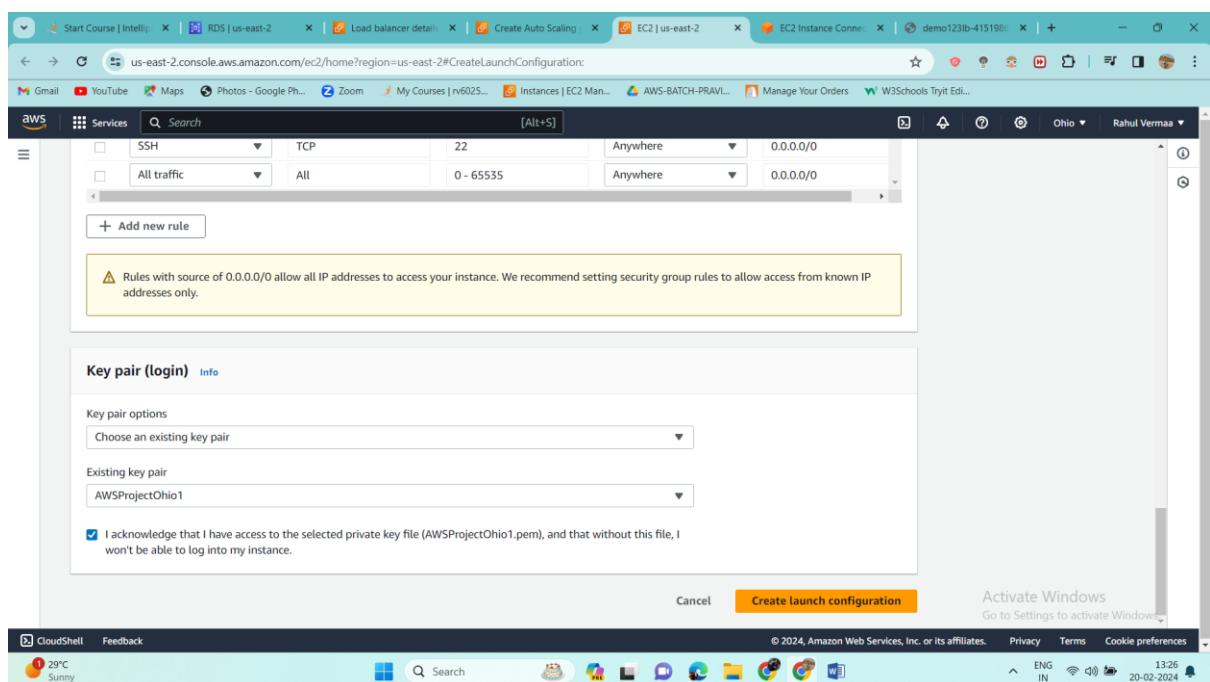


The screenshot shows the AWS EC2 Instances page. The 'Actions' menu is open, and the 'Create image' option is highlighted. A message at the top of the page says: 'Currently creating AMI ami-07cebac03c2d1555 from instance i-03b04445984341010. Check that the AMI status is 'Available' before deleting the instance or carrying out other actions related to this AMI.' The main area shows the instance table with 'AWS1' selected. The 'Details' tab is selected. On the right, there are sections for Instance summary, IP addresses, and Elastic IP addresses. The status bar at the bottom shows the date as 19-02-2024 and the time as 13:40.

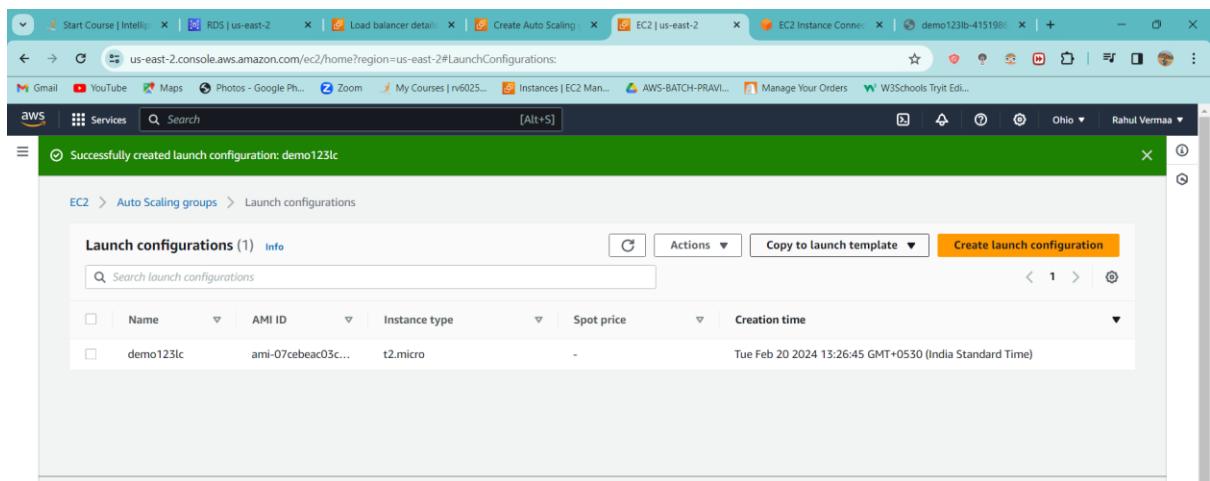
Now we will create launch configuration and will attach our AMI and we have selected instance type as t2. micro



Will select key pair and create our launch configuration



## It's created successfully

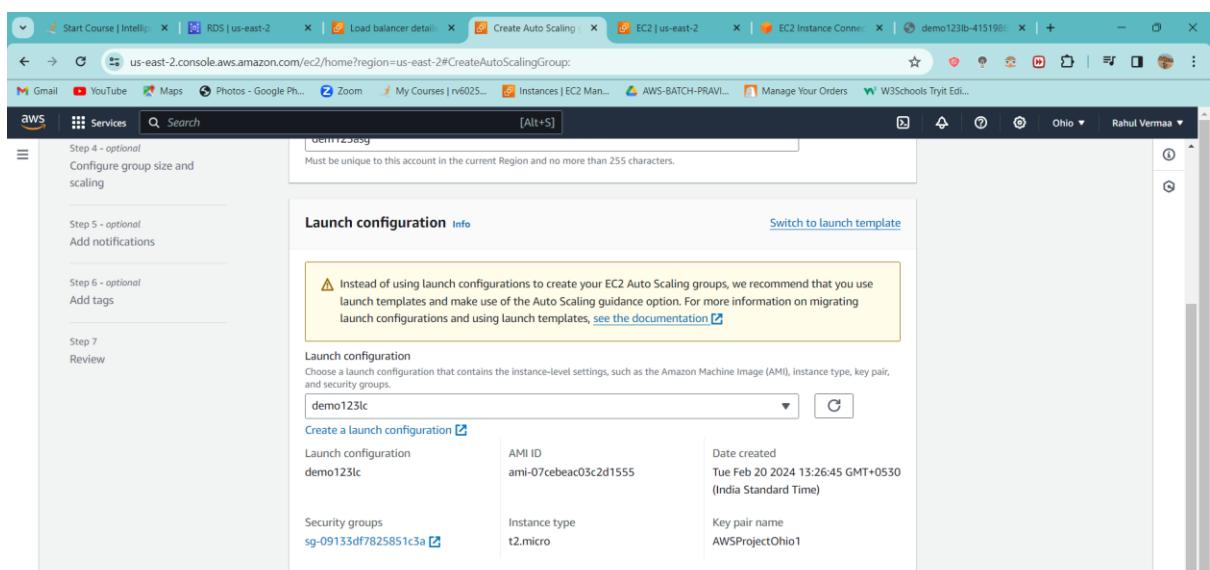


The screenshot shows the AWS EC2 console with the 'Launch configurations' page. A green success message at the top states: 'Successfully created launch configuration: demo123lc'. Below this, a table lists the launch configuration with the following details:

Name	AMI ID	Instance type	Spot price	Creation time
demo123lc	ami-07cebeac03c...	t2.micro	-	Tue Feb 20 2024 13:26:45 GMT+0530 (India Standard Time)

Below the table, a message says 'Select a launch configuration above' with three small icons. The browser status bar at the bottom shows 'CloudShell Feedback' and the date '20-02-2024'.

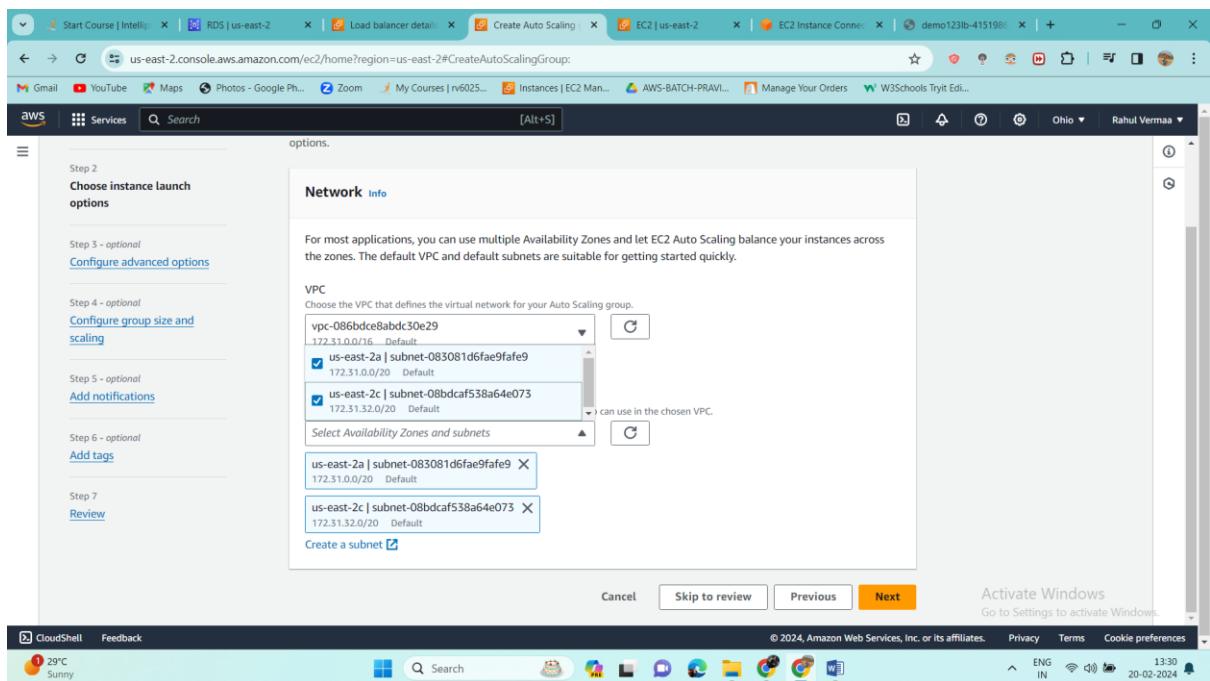
## Now we create our autoscaling group



The screenshot shows the 'Create Auto Scaling Group' wizard, Step 4: 'Configure group size and scaling'. The 'Launch configuration' dropdown is set to 'demo123lc'. Other steps in the wizard are visible on the left: Step 4 (selected), Step 5 (optional), Step 6 (optional), and Step 7 (Review). A note in the center says: 'Instead of using launch configurations to create your EC2 Auto Scaling groups, we recommend that you use launch templates and make use of the Auto Scaling guidance option. For more information on migrating launch configurations and using launch templates, see the documentation'.

At the bottom right, there are 'Cancel' and 'Next' buttons, and the browser status bar shows 'CloudShell Feedback' and the date '20-02-2024'.

## Select vpc and subnets



Step 2  
Choose instance launch options

Step 3 - optional  
[Configure advanced options](#)

Step 4 - optional  
[Configure group size and scaling](#)

Step 5 - optional  
[Add notifications](#)

Step 6 - optional  
[Add tags](#)

Step 7  
[Review](#)

**Network** Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-086bdc8abdc30e29  
172.31.0.0/16 Default

us-east-2a | subnet-083081d6fae9fafe9  
172.31.0.0/20 Default

us-east-2c | subnet-08bdcaf538a64e073  
172.31.32.0/20 Default

Select Availability Zones and subnets

us-east-2a | subnet-083081d6fae9fafe9  
172.31.0.0/20 Default

us-east-2c | subnet-08bdcaf538a64e073  
172.31.32.0/20 Default

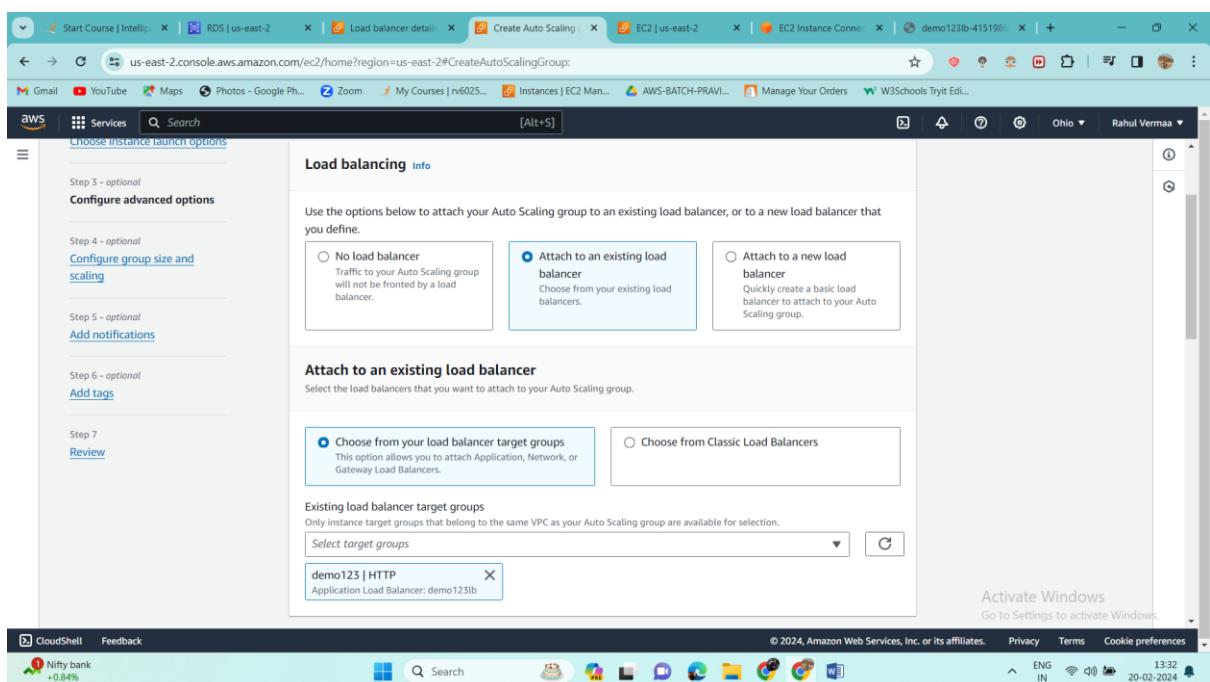
Create a subnet

Cancel

Activate Windows  
Go to Settings to activate Windows.

CloudShell Feedback 29°C Sunny 13:30 20-02-2024 ENG IN 13:30 20-02-2024

## Now attach load balancer



Step 3 - optional  
[Configure advanced options](#)

Step 4 - optional  
[Configure group size and scaling](#)

Step 5 - optional  
[Add notifications](#)

Step 6 - optional  
[Add tags](#)

Step 7  
[Review](#)

**Load balancing** Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer  
Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer  
Choose from your existing load balancers.

Attach to a new load balancer  
Quickly create a basic load balancer to attach to your Auto Scaling group.

**Attach to an existing load balancer**

Select the load balancers that you want to attach to your Auto Scaling group.

Choose from your load balancer target groups  
This option allows you to attach Application, Network, or Gateway Load Balancers.

Choose from Classic Load Balancers

Existing load balancer target groups  
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

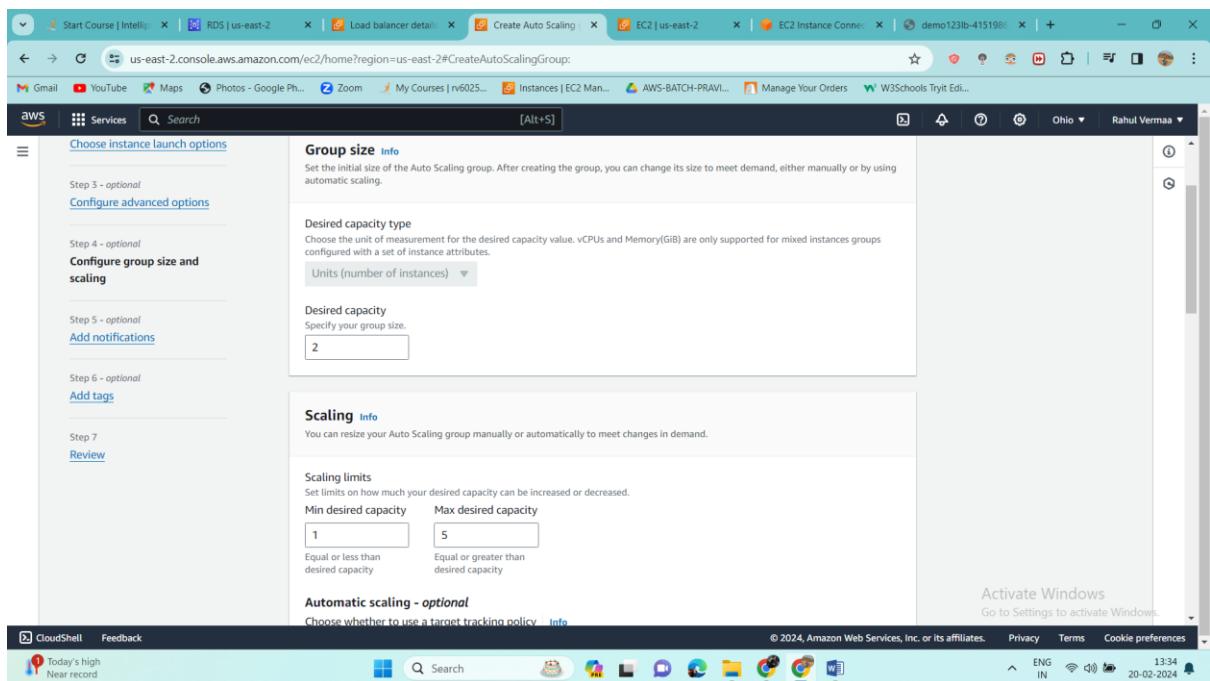
Select target groups

demo123 | HTTP  
Application Load Balancer: demo123lb

Activate Windows  
Go to Settings to activate Windows.

CloudShell Feedback Nifty bank +0.84% 13:32 20-02-2024 ENG IN 13:32 20-02-2024

## We have to give desired capacity requirements



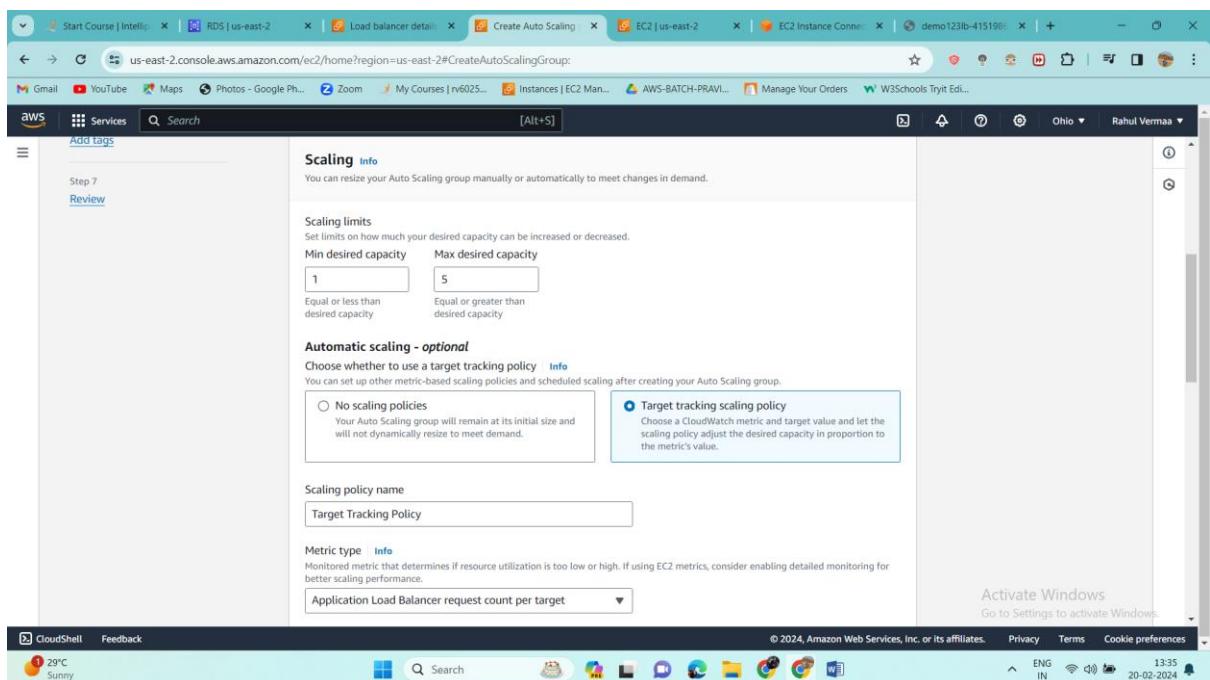
Desired capacity: 2

Min desired capacity: 1

Max desired capacity: 5

Automatic scaling - optional: Target tracking scaling policy

## Scaling



Desired capacity: 2

Min desired capacity: 1

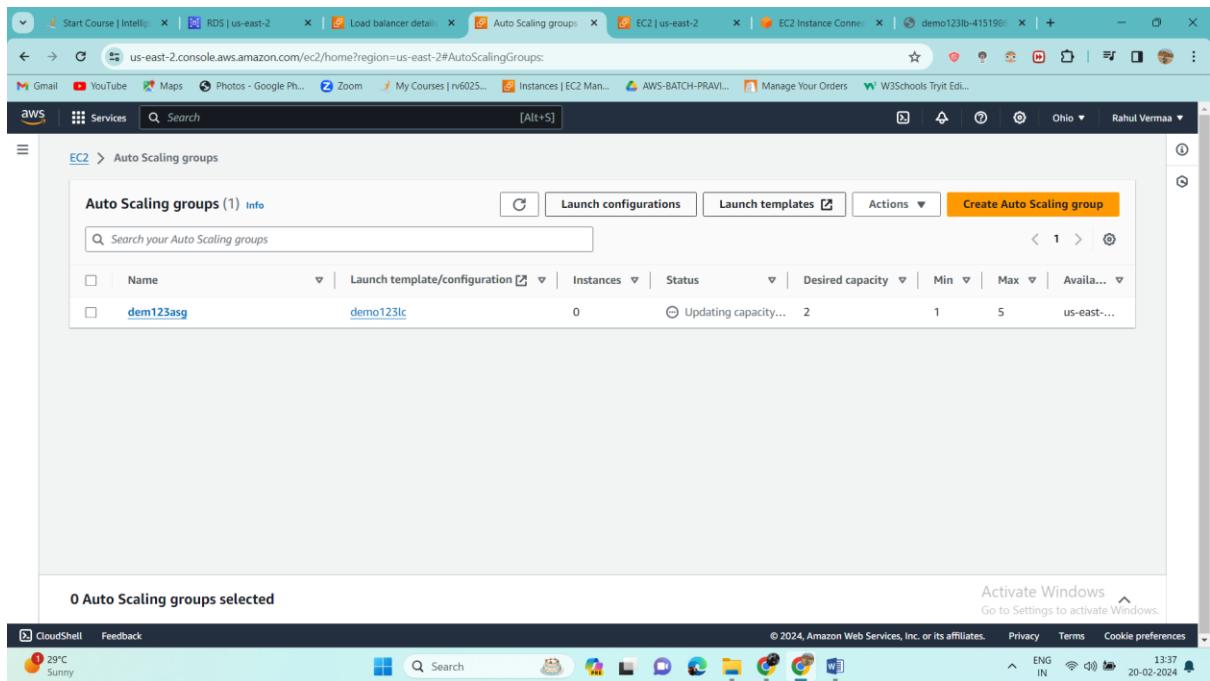
Max desired capacity: 5

Automatic scaling - optional: Target tracking scaling policy

Scaling policy name: Target Tracking Policy

Metric type: Application Load Balancer request count per target

And it's done

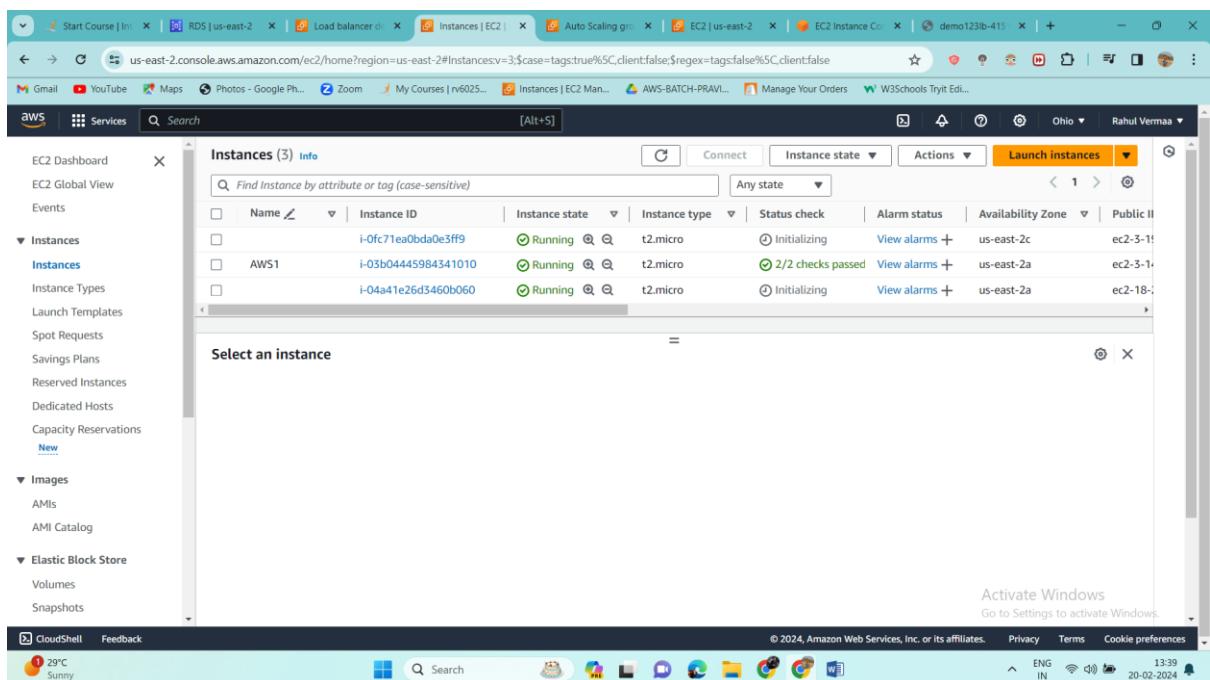


Auto Scaling groups (1) [Info](#)

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Available
demo123asg	demo123lc	0	Updating capacity...	2	1	5	us-east-2

0 Auto Scaling groups selected

And our instances is created



Instances (3) [Info](#)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
i-0fc71ea0bda0e3ff9	i-0fc71ea0bda0e3ff9	Running	t2.micro	Initializing	View alarms +	us-east-2c	ec2-3-11
AWS1	i-03b04445984341010	Running	t2.micro	2/2 checks passed	View alarms +	us-east-2a	ec2-3-1
	i-04a41e26d3460b060	Running	t2.micro	Initializing	View alarms +	us-east-2a	ec2-18-2

Select an instance

The screenshot shows the AWS EC2 Auto Scaling Groups console. The URL in the browser is <https://us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#AutoScalingGroupDetails:id=dem123asg;view=instanceManagement>. The page displays the 'dem123asg' auto scaling group. The 'Instance management' tab is selected. The 'Instances (2)' section shows two instances:   
1. Instance ID: i-04a41e26d3460b060, Lifecycle: Terminating, Instance type: t2.micro, Launch template: demo123lc, Availability zone: us-east-2a, Health status: Healthy.   
2. Instance ID: i-0fc71ea0bda0e3ff9, Lifecycle: InService, Instance type: t2.micro, Launch template: demo123lc, Availability zone: us-east-2c, Health status: Healthy.   
The 'Lifecycle hooks (0)' section shows a message: 'No lifecycle hooks are currently configured.' The bottom of the page includes a weather widget (29°C, Sunny), a search bar, and a footer with copyright information and navigation links.