

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

Problem Statement:

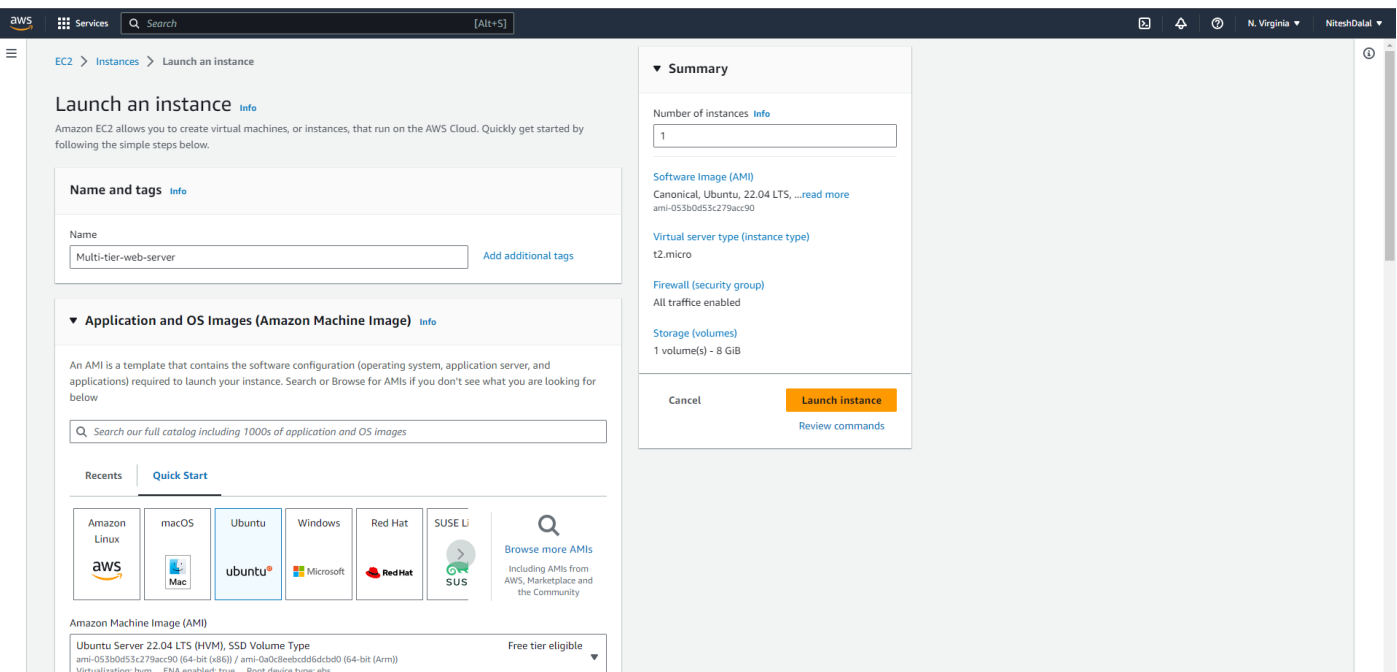
Company ABC wants to move their product to AWS. They have the following things set up right now: 1. MySQL DB 2. Website (PHP) The Company wants high availability on this product, therefore wants Auto Scaling to be enabled on this website.

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

Solution:

1. Go to your EC2 management console and launch an EC2 instance with default VPC and a security group that allows all traffic. Also you can select any region.



2. Once the machine is up and running we will hit some commands on this.

`sudo apt-get update` // to update machine

`sudo apt-get install apache2` // to install ache 2 server

`sudo add-apt-repository -y ppa:ondrej/php` // to install php-mysql

`sudo apt install php5.6 mysql-client php5.6-mysql` // to install php-mysql

3. Now next task is to enable auto scaling and to do that we need to create a image first for our AMI. Then with that image we will create a launch template.
4. Select your EC2 instance and go to Actions tab -> Image and Templates -> Create image.

The screenshot shows the AWS Management Console interface. On the left, there's a navigation menu with categories like 'Instances', 'Images', 'Elastic Block Store', and 'Network & Security'. The main area displays a list of instances, with one instance 'Multi-tier-web-server' selected. Below the list, the 'Instance summary' for 'i-0f90507cb4f63cfac' is shown, detailing its ID, public IPv4 address, instance state (Running), and private IP address. The 'Actions' menu is open, showing various options for managing the instance.

5. Name the image and click create.

The screenshot shows the 'Create image' wizard in the AWS Management Console. The 'Image name' field is filled with 'Multi-tier-web-server-image'. The 'Image description' field is empty. The 'No reboot' checkbox is checked. The 'Instance volumes' section shows a table with columns for Storage type, Device, Snapshot, Size, Volume type, IOPS, Throughput, Delete on termination, and Encrypted. The 'Tags' section shows two options: 'Tag image and snapshots together' (selected) and 'Tag image and snapshots separately'.

6. Once the image is in Available state go ahead and create a launch template. Name it and choose instance type, Key pair & security groups same as your EC2.
7. Also under Application and OS images choose My AMIs -> Owned by me and click on image created by us.

Services

Search

[Alt+S]

N. VirginiaNiteshDatal

EC2 > Launch templates > Create launch template

Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - required

Multi-tier-launch-template

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@', or '-'.

Template version description

A prod webserver for MyApp

Max 255 chars

Auto Scaling guidance

Info

Select this if you intend to use this template with EC2 Auto Scaling

☐ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

Template tags

Source template

Summary

Software Image (AMI)

-

Virtual server type (instance type)

-

Firewall (security group)

-

Storage (volumes)

-

Cancel

Create launch template

Launch template contents

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

Application and OS Images (Amazon Machine Image)

Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Services

Search

[Alt+S]

N. VirginiaNiteshDatal

Application and OS Images (Amazon Machine Image)

Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Recents

My AMIs

Quick Start

☐ Don't include in launch template

☒ Owned by me

☐ Shared with me

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Multi-tier-web-server-image

ami-0bc4b39431a039aaa

2023-08-09T06:51:14.000Z

Virtualization: hvm

ENA enabled: true

Root device type: ebs

Description

-

Architecture

AMI ID

x86_64

ami-0bc4b39431a039aaa

Instance type

Info

Advanced

Instance type

-

Summary

Software Image (AMI)

Multi-tier-web-server-image

ami-0bc4b39431a039aaa

Virtual server type (instance type)

-

Firewall (security group)

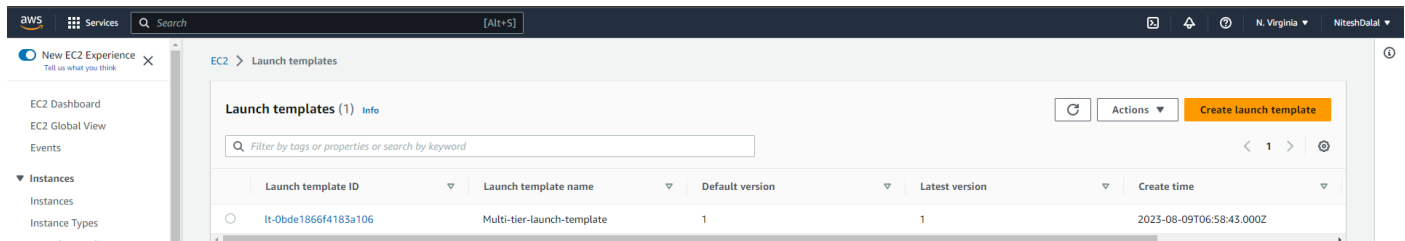
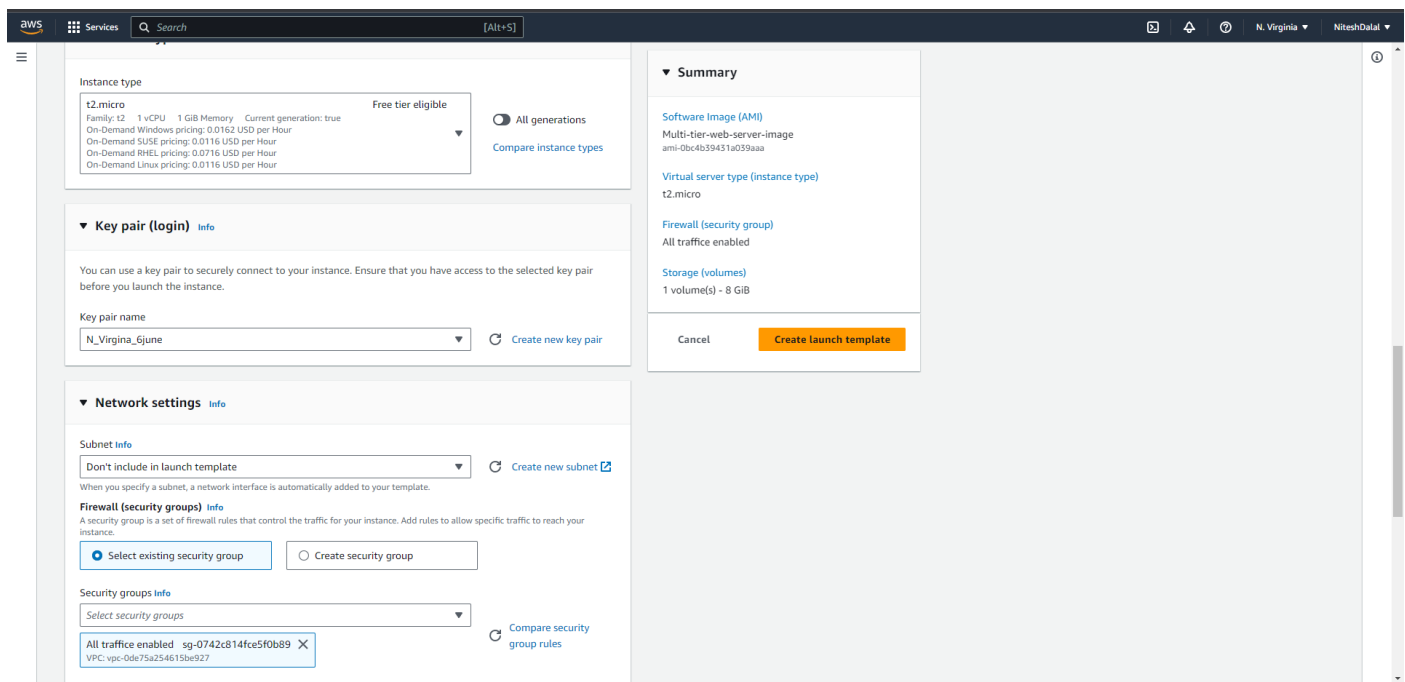
-

Storage (volumes)

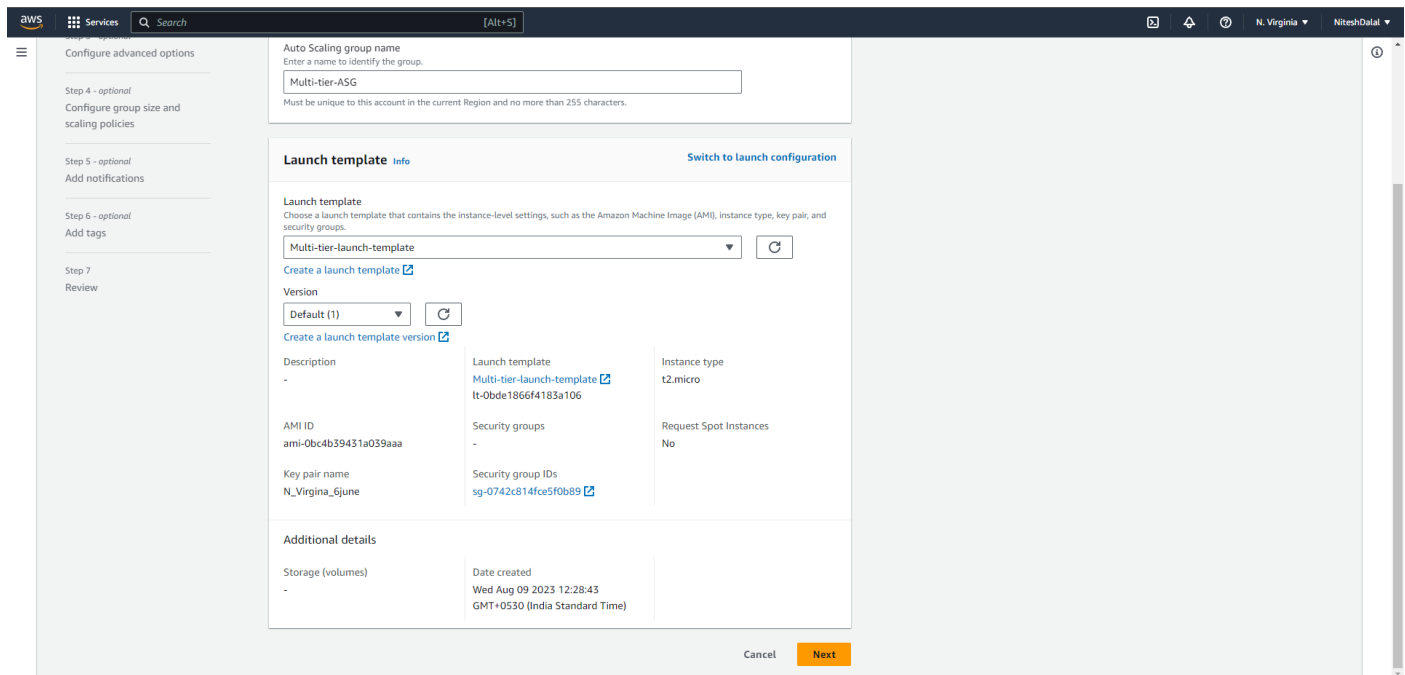
1 volume(s) - 8 GiB

Cancel

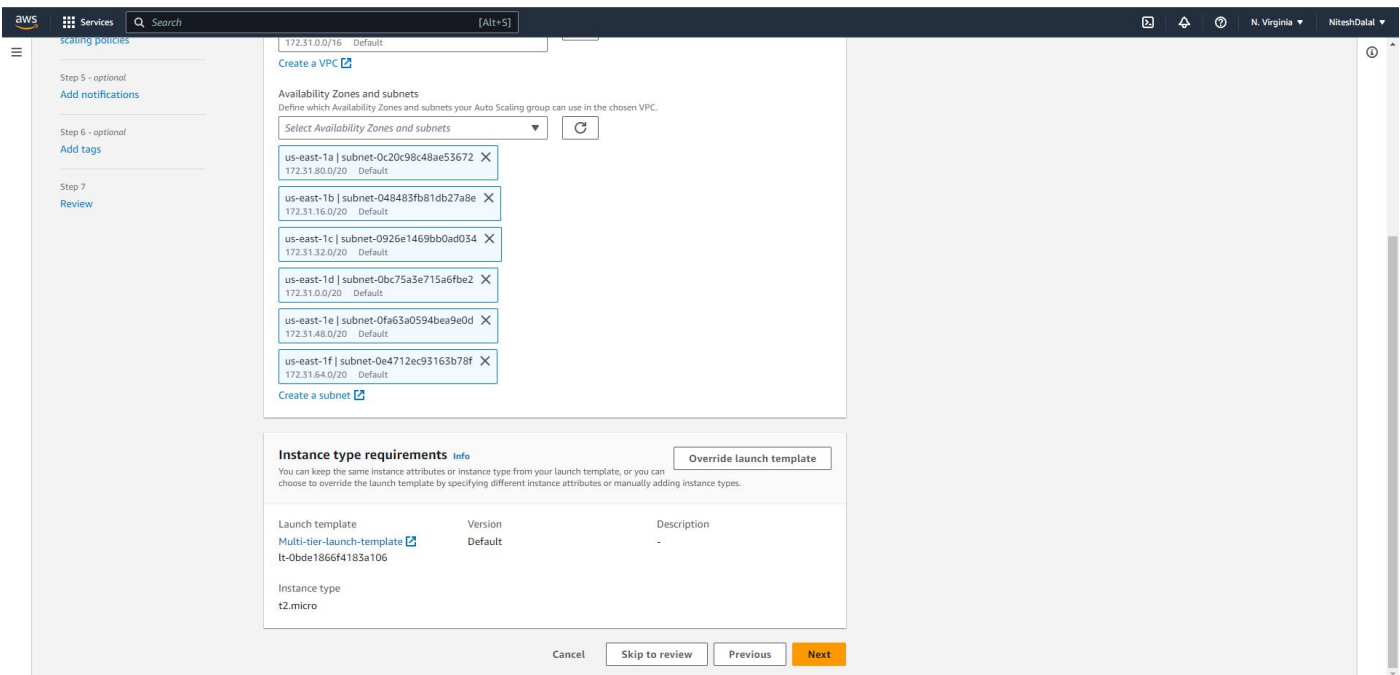
Create launch template



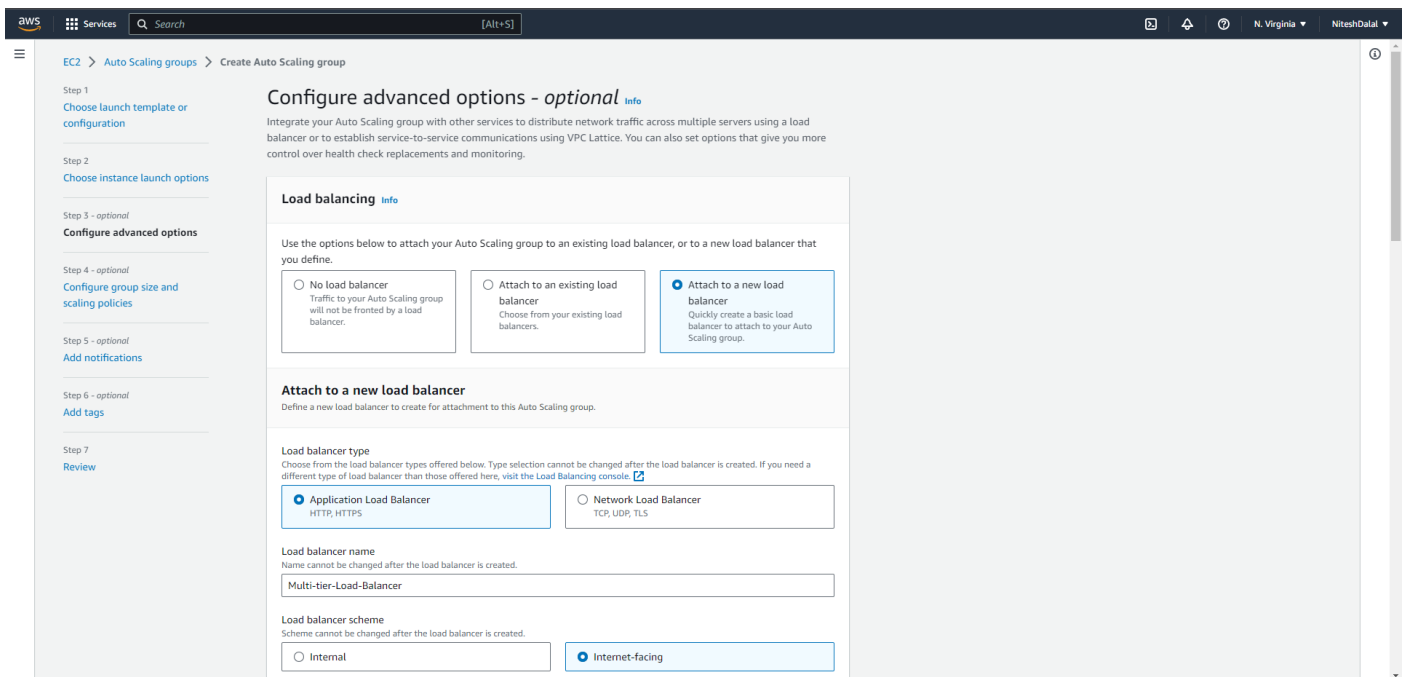
8. Now go to Auto scaling group and create one. Choose your launch template in it.



9. Under next step choose your VPC and availability zones.



10. In next step I am attaching it to a new Load Balancer i.e. Application Load Balancer and Internet facing.



11. Choosing default VPC and AZ's. And for Listeners and routing I am creating a new target group.

Network mapping
Your new load balancer will be created using the same VPC and Availability Zone selections as your Auto Scaling group. You can select different subnets and add subnets from additional Availability Zones.

VPC
vpc-0de75a254615be927

Availability Zones and subnets
You must select a single subnet for each Availability Zone enabled. Only public subnets are available for selection to support DNS resolution.

- ☒ us-east-1e subnet-0fa63a0594bea9e0d
- ☒ us-east-1d subnet-0bc75a3e715a6f6e2
- ☒ us-east-1a subnet-0c20c98c48ae53672
- ☒ us-east-1b subnet-048483fb81db27a8e
- ☒ us-east-1c subnet-0926e1469bb0ad034
- ☒ us-east-1f subnet-0e4712ec93163b78f

Listeners and routing
If you require secure listeners, or multiple listeners, you can configure them from the Load Balancing console after your load balancer is created.

Protocol: HTTP Port: 80 Default routing (forward to): Create a target group

New target group name: Multi-tier-Load-Balancer-TG

Tags - optional
Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add tag (50 remaining)

12. Then as per the question choosing minimum capacity as two and maximum as five. Then click on create ignoring the last two steps.

Choose instance launch options

Step 3 - optional: Configure advanced options

Step 4 - optional: **Configure group size and scaling policies**

Step 5 - optional: Add notifications

Step 6 - optional: Add tags

Step 7: Review

Group size - optional

Specify the size of the Auto Scaling group by changing the desired capacity. You can also specify minimum and maximum capacity limits. Your desired capacity must be within the limit range.

Desired capacity: 2

Minimum capacity: 2

Maximum capacity: 5

Scaling policies - optional

Choose whether to use a scaling policy to dynamically resize your Auto Scaling group to meet changes in demand.

☐ Target tracking scaling policy

☒ None

Instance scale-in protection - optional

Instance scale-in protection

☐ Enable instance scale-in protection

Cancel Skip to review Previous Next

13. See two new servers are automatically triggered.

Instances (3)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4	Public IPv4	Elast
Multi-tier-web-server	i-0f90507cb4f63cfac	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b	ec2-18-209-49-...	18.209.49.242	-
-	i-02fb40277224c0eba	Running	t2.micro	Initializing	No alarms	us-east-1f	ec2-3-235-141-...	3.235.141.29	-
-	i-0bfe1d2ee21851a4c	Running	t2.micro	Initializing	No alarms	us-east-1d	ec2-3-92-18-24-...	3.92.18.247	-

14. Now next task is to change the default webpage of our apache web server. To do that we will open public IP of our instance and navigate to the location of default webpage file i.e. /var/www/html/index.html

15. We are going to remove that file with command `sudo rm index.html` and create a new file there as `index.php`.

```

Last login: Wed Aug  9 06:42:42 2023 from 18.206.107.28
ubuntu@ip-172-31-28-199:~$ cd /var/www/html
ubuntu@ip-172-31-28-199:~$ cd /var/www/html$ ls
index.html
ubuntu@ip-172-31-28-199:~$ cd /var/www/html$ sudo rm index.html
ubuntu@ip-172-31-28-199:~$ cd /var/www/html$ ls
ubuntu@ip-172-31-28-199:~$ cd /var/www/html$

```

16. Created a new index.php with code provided by intellipat for sample website. Save the file with CTRL+S, CTRL+X.

```

GNU nano 6.2 index.php
<td colspan="4">
  <form method="post">
    <div class="form-group" action="post">
      <label for="firstname">Name:</label>
      <input type="text" class="form-control" name="firstname">
    </div>
    <div class="form-group">
      <label for="email">Email:</label>
      <input type="text" class="form-control" name="email">
    </div>
    <div class="form-group">
      <input type="submit" class="btn btn-success">Submit</button>
    </div>
  </form></td>
</tr>
</table>
</div>
</div>
<?php
$firstname=$_POST['firstname'];
$email=$_POST['email'];
$servername = "intelli.coghw13fheqo.us-east-2.rds.amazonaws.com";
$username = "intel";
$password = "intel123";
$db = "intel";
// Create connection
$conn = new mysqli($servername, $username, $password, $db);

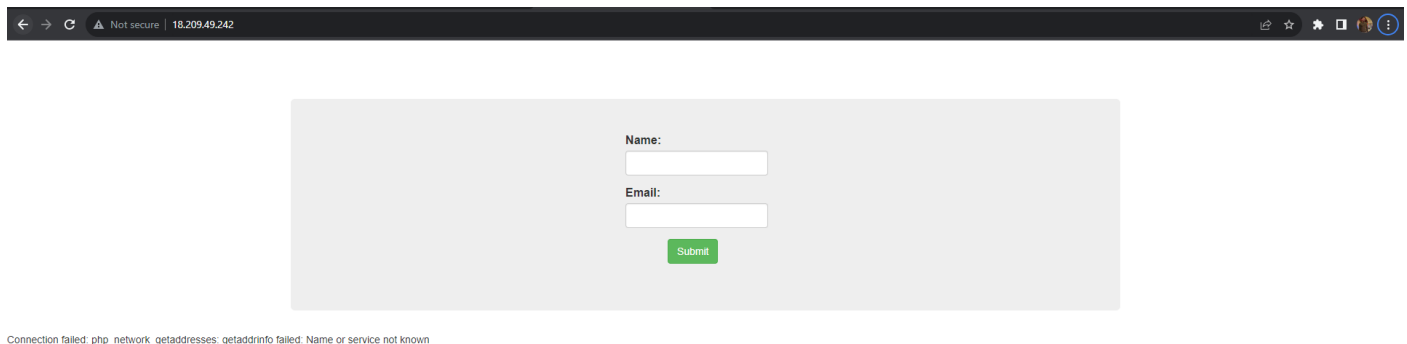
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
if(isset($_POST['firstname']) && isset($_POST['email'])){
    $sql = "INSERT INTO data (firstname,email)
VALUES ('".$_POST['firstname']."','".$_POST['email']."')";

    if ($conn->query($sql) === TRUE) {
        echo "New record created successfully";
    } else {
        echo "Error: " . $sql . "<br>" . $conn->error;
    }
}

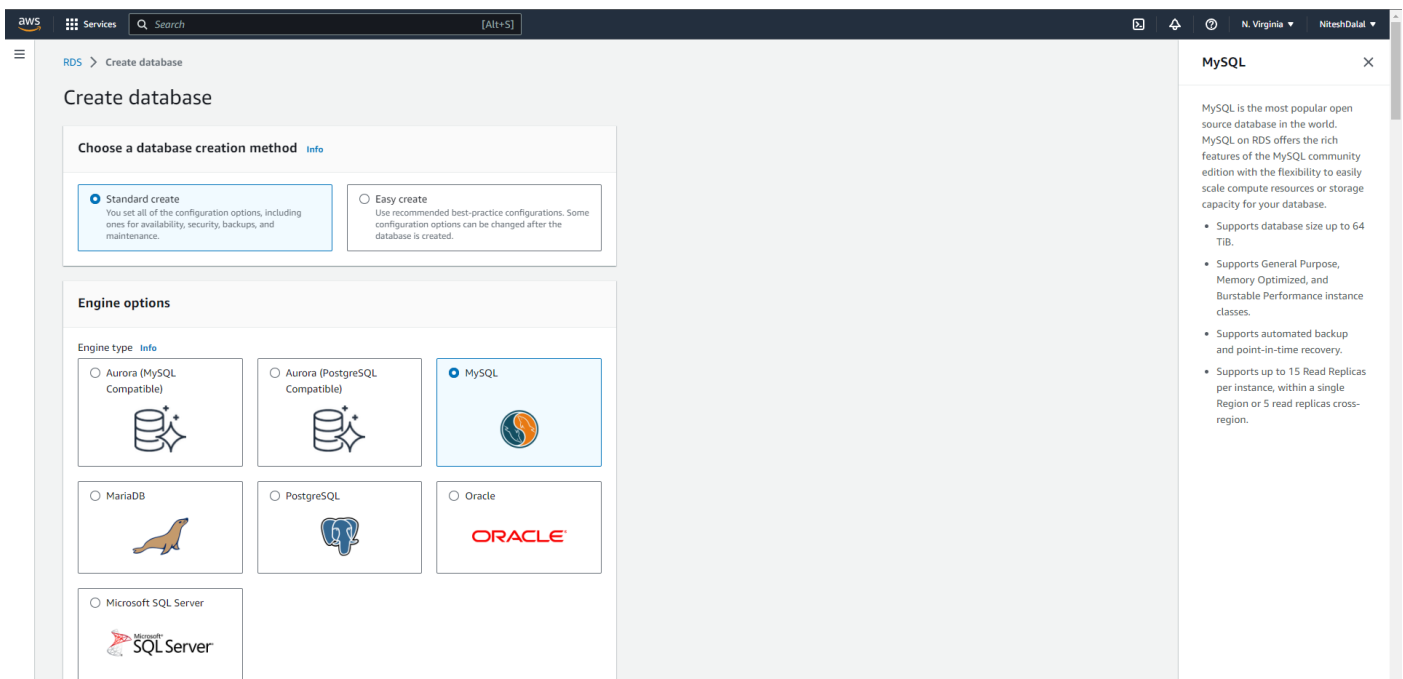
$conn->close();
?>
</body>
</html>

```

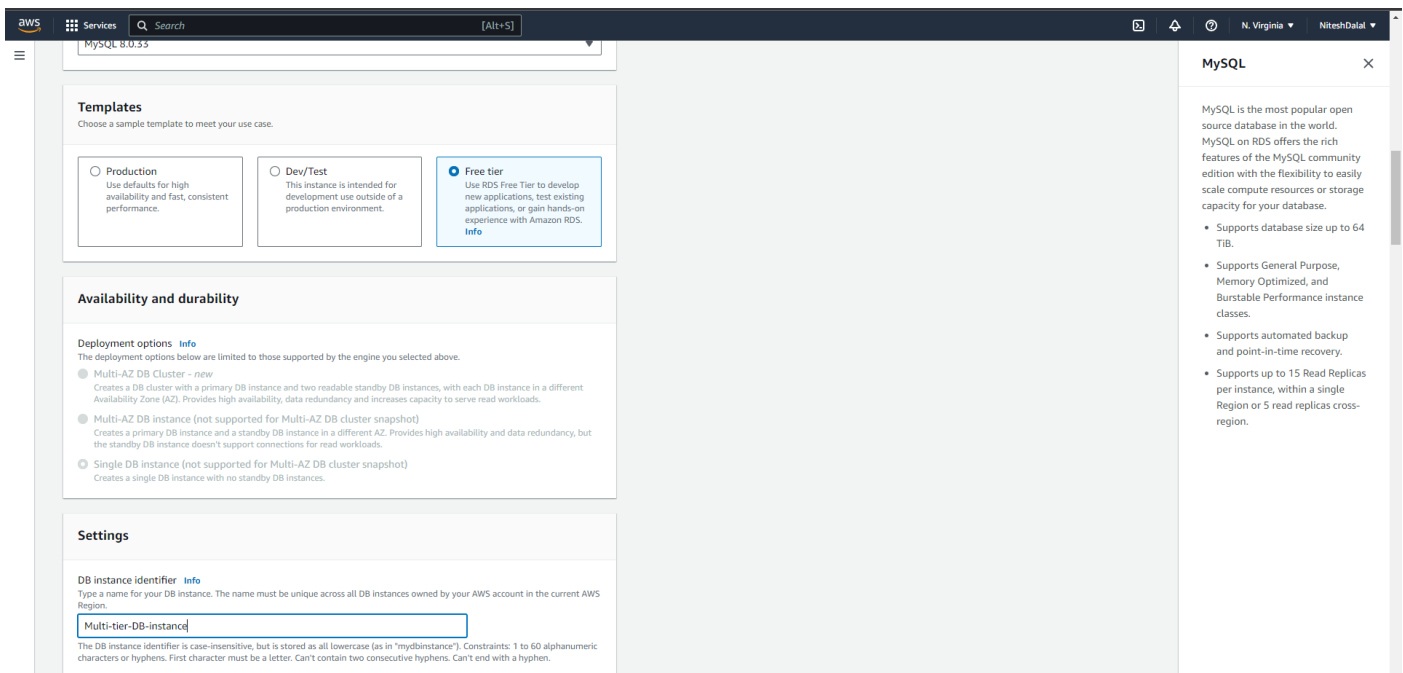
17. After saving your website will look like this.



18. It is showing connection failed error because we haven't create a RDS database yet. So, lets create it. Go with standard create and type as MYSQL.



19. Choose template as free tier and name your DB instance.



20. Give you DB a master username and password for credentialing purpose.

Credentials Settings

Master username [Info](#)
Type a login ID for the master user of your DB instance.
admin
1 to 16 alphanumeric characters. The first character must be a letter.

☐ **Manage master credentials in AWS Secrets Manager**
Manage master user credentials in Secrets Manager. RDS can generate a password for you and manage it throughout its lifecycle.

Master password [Info](#)
Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), ' (single quote), " (double quote) and @ (at sign).
Confirm master password [Info](#)

Instance configuration
The DB instance configuration options below are limited to those supported by the engine that you selected above.

Amazon RDS Optimized Writes - new [Info](#)
☐ Show instance classes that support Amazon RDS Optimized Writes

DB instance class [Info](#)
☐ Standard classes (includes m classes)
☐ Memory optimized classes (includes r and x classes)
☒ **Burstable classes (includes t classes)**

21. Instance configuration with instance class as Burstable classes and storage as 20 GB. Also storage auto calling turned off to avoid cost.

Storage

Storage type [Info](#)
General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage [Info](#)
20 GiB
The minimum value is 20 GiB and the maximum value is 6,144 GiB

Storage autoscaling [Info](#)
Provides dynamic scaling support for your database's storage based on your application's needs.
☐ **Enable storage autoscaling**
Enabling this feature will allow the storage to increase after the specified threshold is exceeded.

22. Under your Connectivity options choose to Connect with an EC2 compute resource and choose your instance.

Compute resource
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

☐ Don't connect to an EC2 compute resource
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

☒ **Connect to an EC2 compute resource**
Set up a connection to an EC2 compute resource for this database.

EC2 instance Info
Choose the EC2 instance to add as the compute resource for this database. A VPC security group is added to this EC2 instance. A VPC security group is also added to the database with an inbound rule that allows the EC2 instance to access the database.

i-0f90507cb4f63cfac
Multi-tier-web-server

Some VPC settings can't be changed when a compute resource is added
Adding an EC2 compute resource automatically selects the VPC, DB subnet group, and public access settings for this database. To allow the EC2 instance to access the database, a VPC security group rds-ec2-X is added to the database and another called ec2-rds-X to the EC2 instance. You can remove the new security group for the database only by removing the compute resource.

Virtual private cloud (VPC) Info
Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

Default VPC (vpc-0de75a254615be927)
11 Subnets, 6 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

DB subnet group Info
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

☐ Choose existing
Choose existing DB subnet group

☒ **Automatic setup**
RDS creates a new subnet group for you or reuses an existing subnet group

DB subnet group name
rds-ec2-db-subnet-group-1

23. Choose security group and then click on create database.

Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

☐ Choose existing
Choose existing DB subnet group

☒ **Automatic setup**
RDS creates a new subnet group for you or reuses an existing subnet group

DB subnet group name
rds-ec2-db-subnet-group-1
Existing DB subnet group reused.

Public access Info
☒ **No**
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

VPC security group (firewall) Info
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

☒ **Choose existing**
Choose existing VPC security groups

☐ Create new
Create new VPC security group

Additional VPC security group
Choose one or more options

All traffic enabled

Amazon RDS will add a new VPC security group rds-ec2-2 to allow connectivity with your compute resource.

Availability Zone Info
us-east-1b

Certificate authority - optional Info
Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database.

24. Now we are going to connect to our database using command `mysql -h "DB endpoint-name" -u "Master-username" -p` and hit enter. Then enter the password as well.

25. Then we will be creating a database with name intel by hitting command: `create database intel;`

26. Then we will hit command: `use intel; //` to switch to our database and create a table over there.

27. Now we will hit command `create table data(firstname varchar(50), email varchar(50));`

```

ubuntu@ip-172-31-28-199:/var/www/html$ mysql -h multi-tier-db-instance.c3bds9nqaese.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: 8.0.33 Source distribution

Copyright (c) 2000, 2023, 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> create database intel;
Query OK, 1 row affected (0.00 sec)

mysql> use intel;
Database changed
mysql> create table data(firstname varchar(50), email varchar(50));
Query OK, 0 rows affected (0.03 sec)

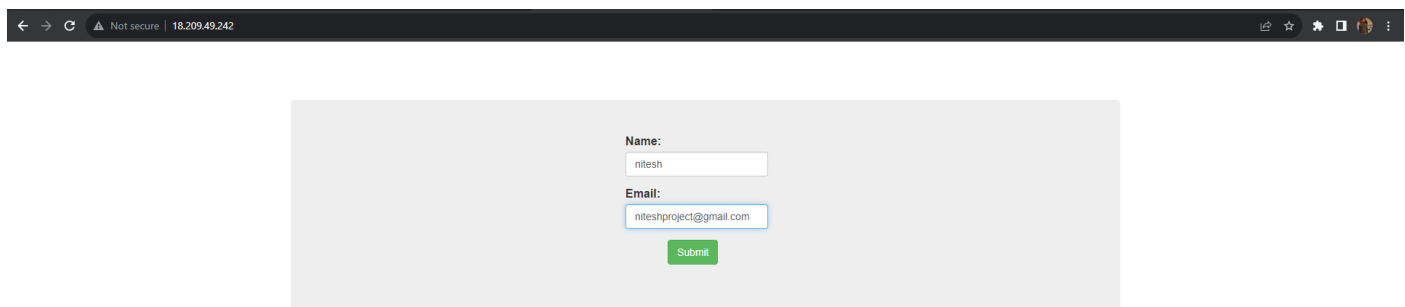
mysql> |

```

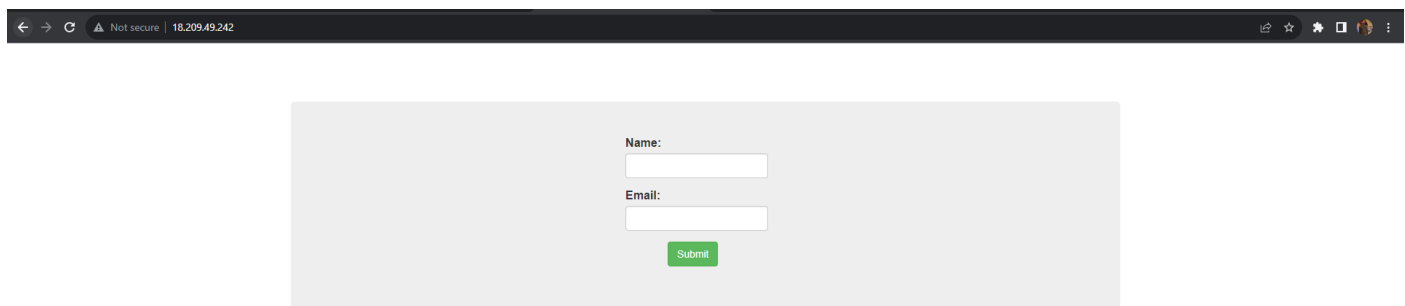
28. Now next step is to edit our index.php file with our DB endpoint, DB name, username and the password.
Using our nano editor.

```
GNU nano 6.2 index.php
<html>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
<body background="images/2.png" style="background-repeat:no-repeat;
background-size: 100% 100%">
<br><br><br><br>
<div class="container">
  <div class="jumbotron vertical-center">
    <table class="grid" cellpadding="0">
      <tr>
        <td colspan="4"><div class="form-group">
          <div class="form-group" action="post">
            <label for="firstname">Name:</label>
            <input type="text" class="form-control" name="firstname">
          </div>
          <div class="form-group">
            <label for="email">Email:</label>
            <input type="text" class="form-control" name="email">
          </div>
          <div class="form-group">
            <button type="submit" class="btn btn-success">Submit</button>
          </div>
        </td>
      </tr>
    </table>
  </div>
</div>
<?php
$firstname=$_POST['firstname'];
$email=$_POST['email'];
$servername = "multi-tier-db-instance.c3bds9nqaese.us-east-1.rds.amazonaws.com";
$username = "admin";
$password = "intel123";
$db = "intel";
// Create connection
$conn = new mysqli($servername, $username, $password, $db);
// Check connection
if ($conn->connect_error) {
  die("Connection failed: " . $conn->connect_error);
}
if(isset($_POST['firstname']) && isset($_POST['email'])) {
  $sql = "INSERT INTO data (firstname,email)
VALUES ('".$_POST['firstname']."','".$_POST['email']."')";
  if ($conn->query($sql) == TRUE) {
    echo "New record created successfully";
  }
}
```

29. Now go back to public IP of our instance and try giving some inputs.



30. See message pop up a new record created successfully.



New record created successfully

31. Now let's check it from the database end. For that we will again use command. "use intel;" & "select * from data;"

```
ubuntu@ip-172-31-28-199: /v...
Server version: 8.0.33 Source distribution

Copyright (c) 2000, 2023, 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> create database intel;
Query OK, 1 row affected (0.00 sec)

mysql> use intel;
Database changed
mysql> create table data(firstname varchar(50), email varchar(50));
Query OK, 0 rows affected (0.03 sec)

mysql> exit
Bye
ubuntu@ip-172-31-28-199: /var/www/html$ cd /var/www/html
-bash: cd /var/www/html: No such file or directory
ubuntu@ip-172-31-28-199: /var/www/html$ sudo nano index.php
ubuntu@ip-172-31-28-199: /var/www/html$ mysql -h multi-tier-db-instance.c3bds9nqaese.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 21
Server version: 8.0.33 Source distribution

Copyright (c) 2000, 2023, 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> 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> select * from data;
+-----+-----+
| firstname | email |
+-----+-----+
| nitesh   | niteshproject@gmail.com |
+-----+-----+
1 row in set (0.01 sec)

mysql> |
```