

CCT College Dublin

Assessment Cover Page

Module Title:	Cloud Computing Fundamentals
Assessment Title:	Autoscaling and NLB in a cloud-based environment
Lecturer Name:	Michael Weiss
Student Full Name:	Muhammad Sajjad Haider
Student Number:	2021384
Assessment Due Date:	Sunday 11 December 23:59
Date of Submission:	Sunday 11 December

Table of Contents

Table of Contents.....	1
Part 1: AWS Bucket	2
Part 2: VPC in one region with two availability zones.....	9
Part 3: AWS VPC diagram.....	12
Part 4: Load Balancer	13
Part 5: Auto Scaling Group.....	21
For Extra Marks	31
Part 6: Amazon Linux VM instance	34
Challenge Task 2:.....	41
Reference	44
Challenge Task 3:.....	45
Reference	51
Challenge Task 4:.....	52
Reference	57

Part 1: AWS Bucket

- Go to **Services**, Select **Storage** and navigate **S3**.
- In Amazon S3 select **Buckets**.
- Click on **Create bucket**.
- I gave **bucket name** as **ca2-2021384**.
- Make sure **AWS Region** in **US East (N. Virginia) us-east-1** as shown below:

The screenshot shows the 'Create bucket' wizard in the AWS S3 console. The top navigation bar shows 'Amazon S3 > Buckets > Create bucket'. The main title is 'Create bucket' with an 'Info' link. A note states 'Buckets are containers for data stored in S3. Learn more' with a link icon. The 'General configuration' section contains fields for 'Bucket name' (set to 'ca2-2021384') and 'AWS Region' (set to 'US East (N. Virginia) us-east-1'). Below these, there's a note about 'Copy settings from existing bucket - optional' and a 'Choose bucket' button. The bottom of the screen shows a progress bar.

- **Untick** the option **Block all public access**.
- Then, **tick** the **acknowledgment** below:

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

⚠ Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

I acknowledge that the current settings might result in this bucket and the objects within becoming public.

- **Bucket Versioning** should be **enabled**.
- **Tags** are optional but I gave **Key** as **Departments** and **Value** as **Sales**.
- **Default Encryption** should also be **enabled**.
- **Encryption key type** should set to first option **SSE-S3**.

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both user actions and application failures. [Learn more](#)

Bucket Versioning

Disable
 Enable

Tags (1) - optional
You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

Key	Value - optional	Remove
Departments	Sales	Remove

[Add tag](#)

Default encryption

Automatically encrypt new objects stored in this bucket. [Learn more](#)

Server-side encryption

Disable
 Enable

Encryption key type
To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

Amazon S3-managed keys (SSE-S3)
An encryption key that Amazon S3 creates, manages, and uses for you. [Learn more](#)

AWS Key Management Service key (SSE-KMS)
An encryption key protected by AWS Key Management Service (AWS KMS). [Learn more](#)

- **Advanced settings** remain **same** and **click on Create bucket**.

AWS Key Management Service key (SSE-KMS)
An encryption key protected by AWS Key Management Service (AWS KMS). Learn more [\[?\]](#)

Advanced settings

Object Lock
Store objects using a write-once-read-many (WORM) model to help you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely. [Learn more \[?\]](#)

Disable
 Enable
Permanently allows objects in this bucket to be locked. Additional Object Lock configuration is required in bucket details after bucket creation to protect objects in this bucket from being deleted or overwritten.

Object Lock works only in versioned buckets. Enabling Object Lock automatically enables Bucket Versioning.

After creating the bucket you can upload files and folders to the bucket, and configure additional bucket settings.

[Cancel](#) [Create bucket](#)

- **Ca2-2021384** bucket has successfully **created** as shown in shot below:

Buckets (2) [Info](#)

Buckets are containers for data stored in S3. [Learn more \[?\]](#)

Name	AWS Region	Access	Creation date
ca2-2021384	US East (N. Virginia) us-east-1	Objects can be public	November 29, 2022, 05:05:39 (UTC+00:00)

- Open the bucket **ca2-2021384**.
- Click on **Upload**.

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#)

Access Points

Objects (0)
Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) [\[?\]](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more \[?\]](#)

[C](#) [Copy S3 URI](#) [Copy URL](#) [Download](#)
[Open](#) [Delete](#) [Actions](#) [Create folder](#)

[Upload](#)

[Find objects by prefix](#) [Show versions](#)

Name	Type	Last modified	Size	Storage class
No objects You don't have any objects in this bucket.				

[Upload](#)

- Inside Upload, click on **Add folder**.

Upload [Info](#)

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add files**, or **Add folders**.

Files and folders (0)	Remove	Add files	Add folder
All files and folders in this table will be uploaded.			
<input type="text"/> Find by name	< 1 >		

- Find the **Digitech html page** from the device and **upload it**.

All files and folders in this table will be uploaded.

<input type="checkbox"/>	Name	Folder	Type	Size
<input type="checkbox"/>	index.html	Digitech_Web-main/	text/html	759.0 B

- Scroll down, before clicking on **upload**, select **Intelligent-Tiering** form **properties**.

Properties
Specify storage class, encryption settings, tags, and more.

Storage class
Amazon S3 offers a range of storage classes designed for different use cases. [Learn more](#) or see [Amazon S3](#)

Storage class	Designed for	Availability Zones
<input type="radio"/> Standard	Frequently accessed data (more than once a month) with milliseconds access	≥ 3
<input checked="" type="radio"/> Intelligent-Tiering	Data with changing or unknown access patterns	≥ 3
<input type="radio"/> Standard-IA	Infrequently accessed data (once a month) with milliseconds access	≥ 3
<input type="radio"/> One Zone-IA	Recreatable, infrequently accessed data (once a month) stored in a single Availability Zone with milliseconds access	1
<input type="radio"/> Glacier Instant	Long-lived archive data accessed once a quarter with instant retrieval in	> 3

- It can be seen **Digitech web page** has successfully uploaded.

Files and folders (1 Total, 759.0 B)						
Name	Folder	Type	Size	Status	Error	
index.html	Digitech_Web-main/	text/html	759.0 B	✓ Succeeded	-	

- Open the **Digitech html**, and **copy its URL**.
- **Search it on google**.
- **It gave error**.

Properties | **Permissions** | **Versions**

Object overview

Owner awslabsc0w4410881t1662750002	S3 URI s3://ca2-2021384/Digitech_Web-main/index.html
AWS Region US East (N. Virginia) us-east-1	Amazon Resource Name (ARN) arn:aws:s3:::ca2-2021384/Digitech_Web-main/index.html
Last modified December 6, 2022, 02:07:46 (UTC+00:00)	Entity tag (Etag) 939360744a5d03d73d97951e70a06266
Size 759.0 B	✓ Object URL Copied
Type html	https://ca2-2021384.s3.amazonaws.com/Digitech_Web-main/index.html
Key Digitech_Web-main/index.html	

- Go back to **Bucket, opened ca2-2021384**.
- Go to **permissions**.
- Click on **Edit Block public access**.
- Untick the **Block all public access**.

Edit Block public access (bucket settings)

Info

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more [\[?\]](#)

Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly created buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

- **Stick in permissions.**
- **Click on Edit Object Ownership.**
- **Enable the access controller ACLs.**
- **Make sure to tick the acknowledgement below.**

Object Ownership

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

ACLs disabled (recommended)

All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

ACLs enabled

Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.



Enabling ACLs turns off the bucket owner enforced setting for Object Ownership

Once the bucket owner enforced setting is turned off, access control lists (ACLs) and their associated permissions are restored. Access to objects that you do not own will be based on ACLs and not the bucket policy.

I acknowledge that ACLs will be restored.

Object Ownership

Bucket owner preferred

If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.

- Go back to **inside the Index.html file.**

- **Edit the access control.**
- **Tick the Read option of Object, in public access.**
- **Tick the acknowledgement below of changes.**
- **Click on Save.**

Access control list (ACL)
Grant basic read/write permissions to AWS accounts. [Learn more](#)

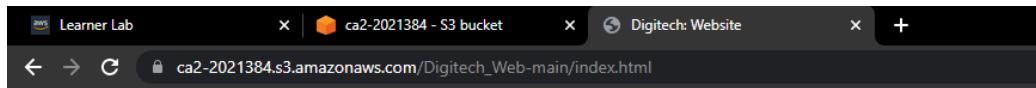
Grantee	Objects	Object ACL
Object owner (your AWS account) Canonical ID: 8028819cd27c773b3d1991204035fe264aaa00881a56de1eb38598b84fed5328	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Read <input checked="" type="checkbox"/> Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	<input checked="" type="checkbox"/> ⚠ Read	<input type="checkbox"/> Read <input type="checkbox"/> Write
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	<input type="checkbox"/> Read	<input type="checkbox"/> Read <input type="checkbox"/> Write

⚠ When you grant access to the Everyone or Authenticated users group grantees, anyone in the world can access this object.
[Learn more](#)

I understand the effects of these changes on this object. /

Access for other AWS accounts

- **Paste the URL of Digitech html page on google and search it.**
- **It can be seen that google is showing successfully the Digitech html page.**



Digitech Company

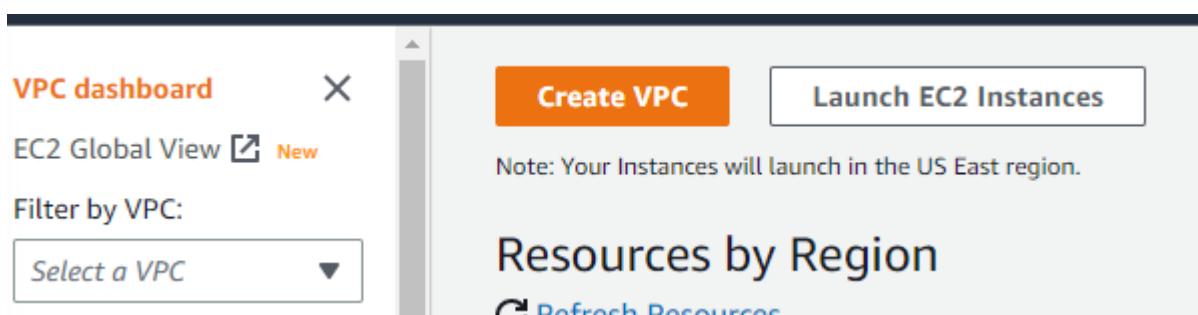


Webmaster : Muhammad Sajjad Haider

Student Number : 2021384

Part 2: VPC in one region with two availability zones

- Go to **Services**, Select **Networking & Content Delivery** and open **VPC**.
- Click on **create VPC**.



- First, I selected **VPC and more**.
- I gave **name tag** as **ca2-2021384-vpc**.
- Select **2** for number of **availability zones**.
- **Tenancy** will be remained **Default**.

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon S3 buckets and Amazon EC2 instances. You can use a VPC to connect your on-premises environment to the AWS Cloud.

VPC settings

Resources to create Info

Create only the VPC resource or the VPC and other networking resources.

 VPC only VPC and more

Name tag auto-generation Info

Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

 Auto-generate

ca2-2021384-vpc

IPv4 CIDR block Info

Determine the starting IP and the size of your VPC using CIDR notation.

10.0.0.0/16

65,536 IPs

IPv6 CIDR block Info

 No IPv6 CIDR block Amazon-provided IPv6 CIDR block

Tenancy Info

Default



Number of Availability Zones (AZs) Info

Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

1

2

3

► Customize AZs

- I chose number of **Public** and **Private subnets** at **2**.
- Nat gateway** at **None**.
- DNS hostnames** and **DNS resolution** both must be **enabled**.
- Click** on **create VPC**.

CUSTOMIZE AZS

Number of public subnets Info
The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

0	2
---	---

Number of private subnets Info
The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

0	2	4
---	---	---

► Customize subnets CIDR blocks

NAT gateways (\$) Info
Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway.

None	In 1 AZ	1 per AZ
------	---------	----------

VPC endpoints Info
Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.

None	S3 Gateway
------	------------

DNS options Info

Enable DNS hostnames

Enable DNS resolution

► Additional tags

Create VPC

- **Ca2-2021384-vpc** has **created** successfully as shown below:

Name	VPC ID	State	IPv4 CIDR
ca2-2021384-vpc-vpc	vpc-00f0082cb1b89c16f	Available	10.0.0.0/16
my-autoscale-vpc-vpc	vpc-00kf4d092e00769172	Available	10.0.0.0/16

- Navigate the **NAT gateway** from left menu of VPC.
- Click on **create NAT gateway**.
- I gave **name** as **my-private-nat-gateway-ca2**.
- Then, I selected the **Private1 subnet**.
- **Connectivity type** on **Private**.
- Click on **create**.

NAT gateway settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.

Connectivity type
Select a connectivity type for the NAT gateway.

Public

Private

Private NAT gateway traffic can't reach the internet.

► Additional settings

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - <i>optional</i>	Remove
<input type="text" value="Name"/>	<input type="text" value="my-private-nat-gateway-ca2"/>	<input type="button" value="Remove"/>

You can add 49 more tags.




- **Private NAT gateway has created successfully.**

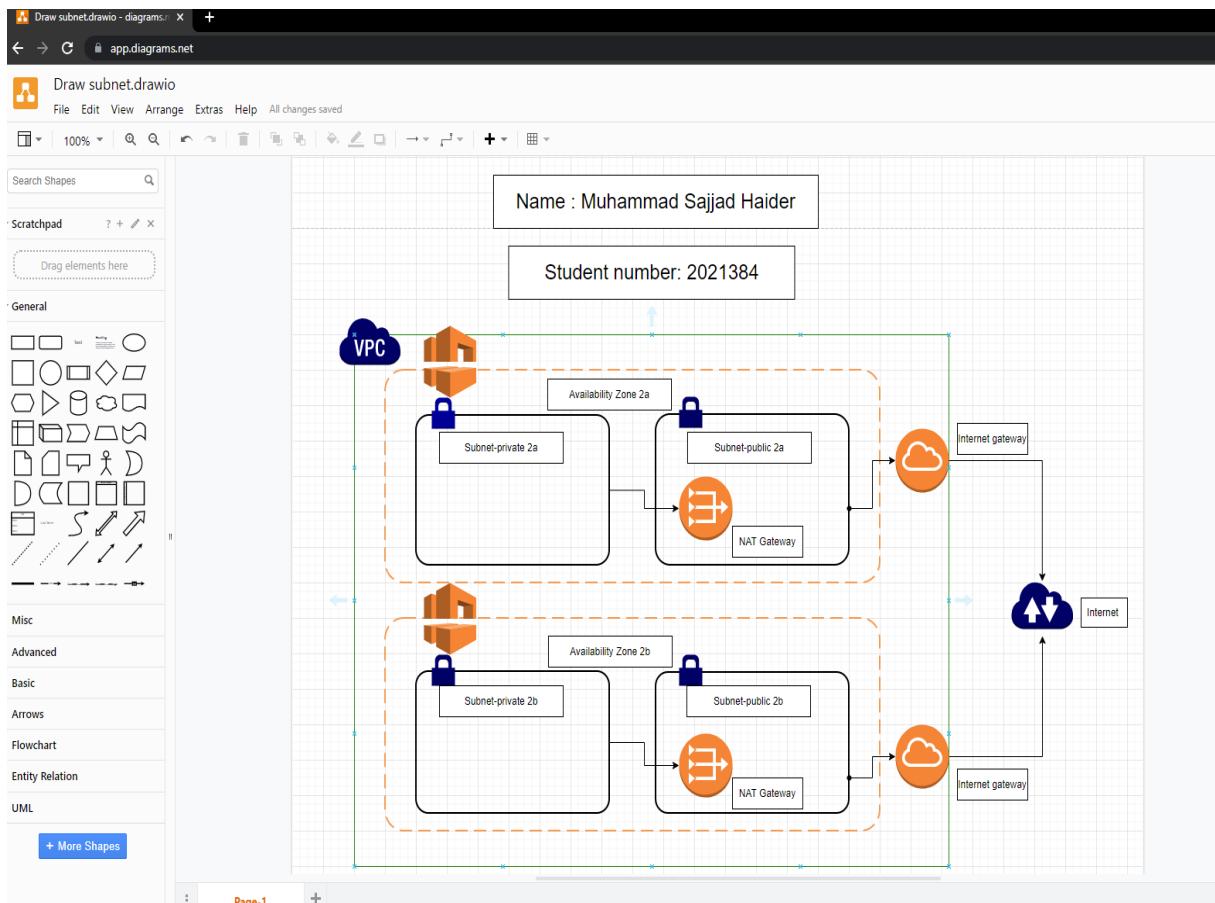
NAT gateways (2) [Info](#)

	Name	NAT gateway ID	Connectivit...	State	State message	Elastic IP address	Primary private ...
<input type="radio"/>	my-private-nat-gateway-ca2	nat-01af8ea7ea7a06e92	Private	Available	-	-	10.0.141.0

Part 3: AWS VPC diagram

- I used **Draw.io** and draw the **AWS VPC** with my **name** and **student number** which I just made in **Part 2**.

- It can be seen that **VPC** in **one region** with **two availability zones**.
- **Each AZ** have **one public subnet** and **one private subnet**.
- There is an **Internet Gateway** for the **public subnet** and a **NAT Gateway** for the **private subnet**.



Part 4: Load Balancer

- For the **load balancer**, First I made **five Linux servers**.
- I made them with the **t2.micro instance type** and **Year-2-cloud-key**.

- First server name is **web-server-ca-1** as shown below:

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name
web-server-ca-1 

Add additional tags

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

Instance type Info

Instance type
t2.micro  Free tier eligible 

Family: t2 1 vCPU 1 GiB Memory
On-Demand Linux pricing: 0.0116 USD per Hour
On-Demand Windows pricing: 0.0162 USD per Hour

Compare instance types

Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*
Year2-Cloud-key  

- I used the **bootstrap** which is given on **Moodle** and I **changed** the statement as “Hello World! This is Web server 1”.

Select a license configuration 

Specify CPU options
The selected instance type does not support CPU options.

Metadata accessible Info
Select 

Metadata version Info
Select 

Metadata response hop limit Info
Select 

Allow tags in metadata Info
Select 

User data Info

```
#!/bin/bash
yum update -y
yum install httpd -y
service httpd start
chkconfig httpd on
cd /var/www/html
echo "<html><h1>Hello World! This is Web server 1 $(hostname -f) </h1></html>" > index.html
```

- I created **second instance** with the name **web-server-ca-2** with same procedure.

Name: web-server-ca-2

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 Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat S

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type
ami-0b0dcb5067f052a63 (64-bit (x86)) / ami-01b5ec3ed8678d8b7 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description: Amazon Linux 2 Kernel 5.10 AMI 2.0.20221103.3 x86_64 HVM gp2

Architecture: AMI ID: ami-0b0dcb5067f052a63 Verified provider

64-bit (x86)

- Here is the **bootstrap** of **Second instance** and I changed the **statement** as "**Hello World! This is Web server 2**".

Specify CPU options
The selected instance type does not support CPU options.

Metadata accessible Info
Select

Metadata version Info
Select

Metadata response hop limit Info
Select

Allow tags in metadata Info
Select

User data Info

```
#!/bin/bash
yum update -y
yum install httpd -y
service httpd start
chkconfig httpd on
cd /var/www/html
echo "<html><h1>Hello World! This is web server 2</h1></html>" > index.html
```

- By following the **same procedure**, I made **five instances** with the **numbering 1 to 5** as shown in the screenshot below:

Instances (6) Info											
Connect Actions Launch instances											
Instance state Public IPv4 DNS Public IPv4 ... Elastic IP											
	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	
<input type="checkbox"/>	web-server-ca-3	i-0361c231631b9dc42	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-184-72-122-118.co...	184.72.122.118	-	
<input type="checkbox"/>	web-server-ca-5	i-0c14dc6ed9e9385a	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-35-174-174-218.co...	35.174.174.218	-	
<input type="checkbox"/>	web-server-ca-4	i-00cb4a61d4631810c	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-100-26-201-90.co...	100.26.201.90	-	
<input type="checkbox"/>	web-server-ca-1	i-0520681c6d1410ed7	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-44-208-28-125.co...	44.208.28.125	-	
<input type="checkbox"/>	web-server-ca-2	i-03c98c1d50463d656	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-3-82-153-114.com...	3.82.153.114	-	

- Then, I created **Target group** with the name **ca2-target-group**, sticking with **instances in target type**.

Choose a target type

Instances

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of Amazon EC2 Auto Scaling to manage and scale your EC2 capacity.

IP addresses

- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same instance.
- Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

Lambda function

- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

Application Load Balancer

- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Target group name
*

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol	Port
<input style="width: 100px; height: 20px; border: none; background-color: #f0f0f0; font-size: small; margin-right: 10px;" type="button" value="HTTP"/>	<input style="width: 100px; height: 20px; border: none; font-size: small;" type="text" value="80"/>

- I also added **tags** with **Key as Departments** and **Value as Sales** but its **optional**.
- Click on **Next**.

▼ **Tags - optional**

Consider adding tags to your target group. Tags enable you to categorize your AWS resources so you can more easily manage them

Key	Value	Remove
<input style="width: 100%; border: none; border-bottom: 1px solid #ccc; height: 20px;" type="text" value="departments"/>	<input style="width: 100%; border: none; border-bottom: 1px solid #ccc; height: 20px;" type="text" value="sales"/>	Remove
Add tag		

You can add up to 49 more tags.

- I selected all five instances in available instances.
- Click on include as pending below option at bottom.

- And click on **create target group**.

Available instances (5/5)

<input checked="" type="checkbox"/>	Instance ID	Name	State	Security groups	Zone	Subnet ID
<input checked="" type="checkbox"/>	i-0361c231631b9dc42	web-server-ca-3	running	launch-wizard-14	us-east-1c	subnet-03c6169da74813
<input checked="" type="checkbox"/>	i-0c14dc6ed99e9385a	web-server-ca-5	running	launch-wizard-16	us-east-1c	subnet-03c6169da74813
<input checked="" type="checkbox"/>	i-00cb4a61d4631810c	web-server-ca-4	running	launch-wizard-15	us-east-1c	subnet-03c6169da74813
<input checked="" type="checkbox"/>	i-0520681c6d1410ed7	web-server-ca-1	running	launch-wizard-12	us-east-1c	subnet-03c6169da74813
<input checked="" type="checkbox"/>	i-03c98c1d50463d656	web-server-ca-2	running	launch-wizard-13	us-east-1c	subnet-03c6169da74813

5 selected

Ports for the selected instances
Ports for routing traffic to the selected instances.

80
1-65535 (separate multiple ports with commas)

Include as pending below

1-65535 (separate multiple ports with commas)

Include as pending below ✓

5 selections are now pending below. Include more or register targets when ready.

Review targets

Targets (5)

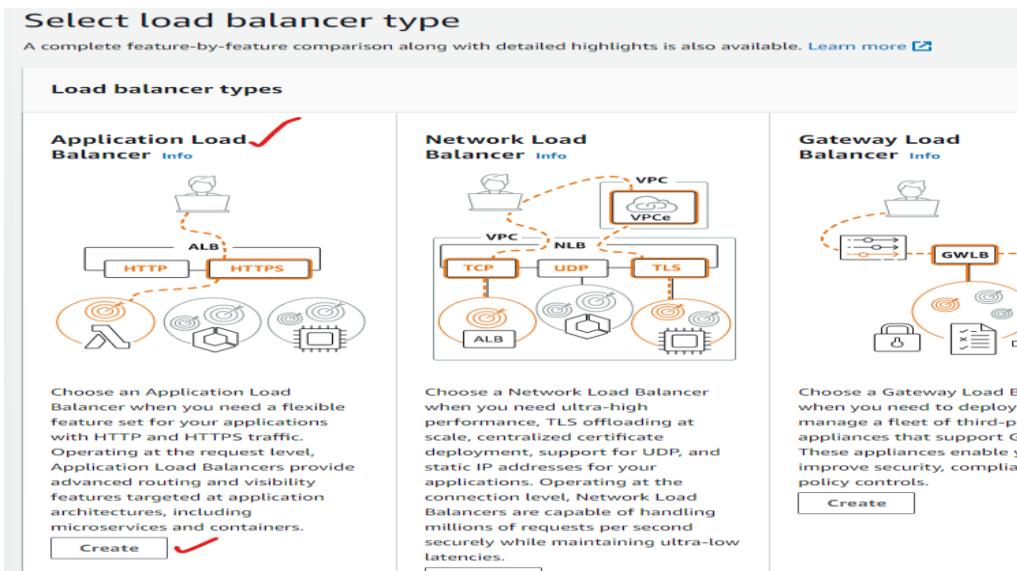
<input checked="" type="checkbox"/>	Health status	Instance ID	Name	Port	State	Secur
X	Pending	i-0361c231631b9dc42	web-server-ca-3	80	running	laun
X	Pending	i-0c14dc6ed99e9385a	web-server-ca-5	80	running	laun
X	Pending	i-00cb4a61d4631810c	web-server-ca-4	80	running	laun
X	Pending	i-0520681c6d1410ed7	web-server-ca-1	80	running	laun
X	Pending	i-03c98c1d50463d656	web-server-ca-2	80	running	laun

Remove all pending

5 pending **Create target group**

- After that, navigate the **load balancer** from the left menu and click on **create load balancer**.
- In **load balancer type**, select **Application Load Balancer**.

- And click on **create**.



- In **Basic configuration**, I gave **load balancer name** as **ca2-2021384-nlb**.
- Stick with **Internet-facing** in **Scheme** and **IPv4** in **IP address type**.
- I selected **Default VPC** in **Network mapping**.

Basic configuration

Load balancer name
Name must be unique within your AWS account and cannot be changed after the load balancer is created.
ca2-2021384-nlb

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme
Scheme cannot be changed after the load balancer is created.

Internet-facing
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)

Internal
An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type
Select the type of IP addresses that your subnets use.

IPv4
Recommended for internal load balancers.

Dualstack
Includes IPv4 and IPv6 addresses.

Network mapping
The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC
Select the virtual private cloud (VPC) for your targets. Only VPCs with an internet gateway are enabled for selection. The VPC cannot be changed after the load balancer is created. To confirm the VPC for your targets, view your target groups.

vpc-0609b8b9b73d5ea21
IPv4: 172.31.0.0/16

- Then, I **created new Security group** with the name **ca2-security-grp** and **Description (Optional)** as **new-nlb-sg**.
- I also set **Inbound rule** with **HTTP**.

- Click next.

Create security group [Info](#)

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details	
Security group name Info	<input type="text" value="ca2-security-grp"/> /
Description Info	<input type="text" value="new-nlb-sg"/> /
VPC Info	<input type="text" value="vpc-0609b8b9b73d5ea21"/> / X

Inbound rules Info /				
Type Info	Protocol Info	Port range Info	Source Info	Description
Custom TCP	TCP	0	Anywhere-IP X /	<input type="text" value="0.0.0.0"/> / X
HTTP	TCP	80	Anywhere-IP X /	<input type="text" value="0.0.0.0"/> / X
... /				

- I selected the new security group named **ca2-security-grp** which I made before and also selected **ca2-target-group** in Default action of Listeners and routing.

Security groups [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer.

Security groups	
Select up to 5 security groups	<input type="button" value="C"/>
Create new security group Create	
<input type="checkbox"/> default sg-0a492651ed64c75cd / X	<input type="checkbox"/> ca2-security-grp sg-02e8efce8040a9665 / X

Listeners and routing [Info](#)

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

Listener HTTP:80		<input type="button" value="Remove"/>
Protocol	Port	Default action Info
HTTP ▼	: 80 1-65535	Forward to ca2-target-group / . HTTP C Create target group /

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

- After clicking on **create**, it can be seen in shot below that **ca2-2021384-nlb** has created successfully.

<input type="checkbox"/>	ca2-2021384-nlb	ca2-2021384-nlb-1909468779.us-east-1.elb.amazonaws.com	Active	vpc-0609b8b9b73d5ea21	6 Availability Zones	application
--------------------------	-----------------	--	--------	-----------------------	----------------------	-------------

- Copy the DNS name of ca2-2021384-nlb and search it into google.

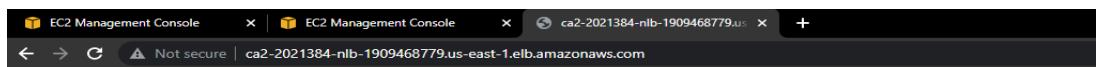
ca2-2021384-nlb

▼ Details

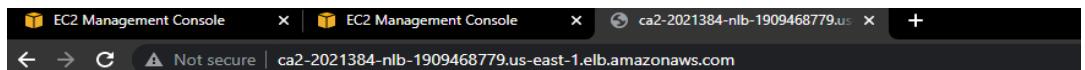
arn:aws:elasticloadbalancing:us-east-1:066970063753:loadbalancer/app/ca2-2021384-nlb/e44036a637323e90

Load balancer type Application Load Balancer	DNS name ca2-2021384-nlb-1909468779.us-east-1.elb.amazonaws.com (A Record)
IP address type IPv4	Scheme Internet-facing

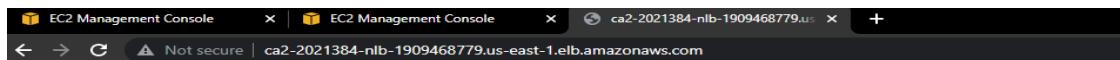
- It can be seen in screenshots below that the **load balancer is working properly.**
- It showed the **page of every instance** by **refreshing the web page** again and again.



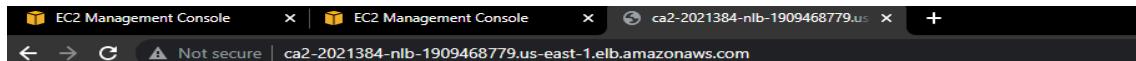
Hello World! This is Web server 1 ip-172-31-90-254.ec2.internal



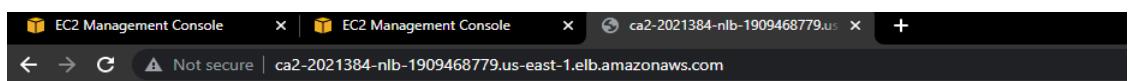
Hello World! This is web server 2 ip-172-31-82-89.ec2.internal



Hello World! This web server 03 ip-172-31-90-122.ec2.internal



Hello World! This is web server 5 ip-172-31-81-154.ec2.internal



Hello World! This is web server 04 ip-172-31-88-68.ec2.internal

Part 5: Auto Scaling Group

- For **Auto scaling**, Firstly I made **template** for it by navigating **Launch Template** from left menu.
- Click on **Launch template**.
- I gave it to **name** as **my-ca2-ASG-template** and write **version1** in **template version** option.
- Make sure to **tick the Auto Scaling guidance**.
- I gave **tag** value as **my-ca2-tamplate**.

Create launch template

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

Launch template name and description

Launch template name - required
my-ca2-ASG-template /

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

Template version description
version1 /

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

Key Info	Value Info	Remove tag
<input type="text"/> Name	<input type="text"/> my-ca2-tamplate	<input type="button"/>

Add tag

49 remaining (Up to 50 tags maximum)

- Click on **Browse more AMIs**.

Application and OS Images (Amazon Machine Image) - required 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 Quick Start

Recently launched Currently in use

Browse more AMIs Including AMIs from AWS, Marketplace and the Community

- I selected the **very first Amazon Linux 2 AMI**.
- And it **showed as selected** below.

All products (47 filtered, 47 unfiltered)

My AMIs (0) AWS Marketplace AMIs (6634) Community AMIs (500)

Created by me AWS & trusted third-party AMIs Published by anyone

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type

ami-0b0dc5067f052a63 (64-bit (x86)) / ami-01b5ec3ed8678d8b7 (64-bit (Arm))

Amazon Linux 2 comes with five years support. It provides Linux kernel 5.10 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is now under maintenance only mode and has been removed from this wizard.

Platform: amazon Root device type: ebs Virtualization: hvm ENA enabled: Yes

Select 64-bit (x86) 64-bit (Arm)

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) - required 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

AMI from catalog Recents Quick Start

Amazon Machine Image (AMI)

amzn2-ami-kernel-5.10-hvm-2.0.20221103.3-x86_64-gp2
ami-0b0dc5067f052a63

Free tier eligible **Verified provider**

Catalog Published Architecture Virtualization Root device type ENA Enabled

Quickstart AMIs 2022-11-14T23:11:49.00Z x86_64 hvm ebs Yes

Browse more AMIs Including AMIs from AWS, Marketplace and the Community

- I used **t2.micro** free tier eligible for this instance.
- Selected **Year2-Cloud-key** for key pair.
- In **Firewall**, I created new security group with name **my-ca2-ASG-SG** and gave **description** as **ASG-ca2-Security-group**.
- I selected default VPC as shown below in screenshot.

Subnet Info
Don't include in launch template

Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Select existing security group Create security group

Security group name - required
my-ca2-ASG-SG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _:-/()#,@[]+=;&:;!\$*

Description - required Info
ASG-ca2-Security-group

VPC - required Info
vpc-0609b8b9b73d5ea21 172.31.0.0/16 (default)

- I have **created two inbound security groups rules**, one with the type on **SSH** and the **other** with the type off **HTTP**.
- I chose **CIDR 0.0.0.0/0** for **both** of them.

VPC - required Info
vpc-0609b8b9b73d5ea21 172.31.0.0/16 (default)

Inbound security groups rules

Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type Info ssh	Protocol Info TCP	Port range Info 22
Source type Info Custom	Source Info Add CIDR, prefix list or security 0.0.0.0/0	Description - optional Info e.g. SSH for admin desktop

Security group rule 2 (TCP, 80, 0.0.0.0/0)

Type Info HTTP	Protocol Info TCP	Port range Info 80
Source type Info Custom	Source Info Add CIDR, prefix list or security 0.0.0.0/0	Description - optional Info e.g. SSH for admin desktop

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

- I have set **Bootstrap** in **additional information** and wrote my **name** and **student number** in it.
- Click on **create**.

The screenshot shows the 'User data' section of a Lambda function configuration. The code is as follows:

```

#!/bin/bash
yum update -y
yum install httpd -y
service httpd start
chkconfig httpd on
cd /var/www/html
echo "<html><h1>Hello World! My name is Muhammad Sajjad Haider and my
Student number is 2021384 $(hostname -f) </h1></html>" > index.html

```

User data has already been base64 encoded

- It can be seen that **my-ca2-ASG-template** has **created**.

Launch templates (2) Info		Actions ▾		Create launch	
	Launch template ID	Launch template name	Default version	Latest version	Create time
○	lt-0a67a5b98f8a41357	my-ca2-ASG-template	1	1	2022-12-03T02:54:32.000Z

- In **EC2**, navigate **Auto scaling groups**.
- Click on **create Auto Scaling group**.
- I gave it to **name** as **my-ca2-ASG**.
- In **Launch template**, I selected the **template** which I have just created with the name **my-ca2-ASG-template**.
- Click **next**.

Choose launch template or configuration Info

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.

Name	
Auto Scaling group name <small>Enter a name to identify the group.</small> <input type="text" value="my-ca2-ASG"/> -	
<small>Must be unique to this account in the current Region and no more than 255 characters.</small>	
Launch template <small>Info</small>	Switch to launch configuration
Launch template <small>Info</small> <small>Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.</small> <input type="text" value="my-ca2-ASG-template"/> -	Create a launch template -
<small>Version</small>	

- In Network, I selected **default VPC** and all **default subnets**.
- Click **Next**.

Network <small>Info</small>	
For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance across the zones. The default VPC and default subnets are suitable for getting started quickly.	
VPC <small>Choose the VPC that defines the virtual network for your Auto Scaling group.</small> <input type="text" value="vpc-0609b8b9b73d5ea21"/> - C 172.31.0.0/16 Default	
Create a VPC -	
Availability Zones and subnets <small>Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.</small> <input type="text" value="Select Availability Zones and subnets"/> - C	
us-east-1a subnet-0bf7b59e7897b940e X 172.31.32.0/20 Default	
us-east-1b subnet-0ad3e2411e00dd99b X 172.31.0.0/20 Default	
us-east-1c subnet-03c616919da748132 X 172.31.80.0/20 Default	
us-east-1d subnet-0b6c1e156cf12c292 X 172.31.16.0/20 Default	
us-east-1e subnet-00573d7cd051d5996 X 172.31.48.0/20 Default	
us-east-1f subnet-0087821b7696f5dbe X 172.31.64.0/20 Default	
Create a subnet -	
Instance type requirements <small>Info</small>	
Override	

- Stick with **NO load balancer** and set **30 seconds** to **Health checks** (Optional).
- Click **Next.**

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

<input checked="" type="radio"/> No load balancer Traffic to your Auto Scaling group will not be fronted by a load balancer.	<input type="radio"/> Attach to an existing load balancer Choose from your existing load balancers.	<input type="radio"/> Attach to a new load balancer Quickly balance traffic between your Auto Scaling group and a new load balancer.
--	---	--

Health checks - optional

Health check type | [Info](#)
EC2 Auto Scaling automatically replaces instances that fail health checks. If you enabled load balancing, health checks in addition to the EC2 health checks that are always enabled.

EC2 ELB

Health check grace period
The amount of time until EC2 Auto Scaling performs the first health check on new instances after they are created.

30 seconds

- I have set **3 Desired capacity**, **2 Minimum capacity** and **5 Maximum capacity**.
- In **Scaling policies** (Optional), I selected **Target tracking scaling policy**.
- I have set **target value 30**.
- I have set **warm up time for instances 30 seconds**.

maximum capacity limits. Your desired capacity must be within the limit range.

Desired capacity	3
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. [Info](#)

<input checked="" type="radio"/> Target tracking scaling policy Choose a desired outcome and leave it to the scaling policy to add and remove capacity as needed to achieve that outcome.	<input type="radio"/> None
---	----------------------------

Scaling policy name
Target Tracking Policy

Metric type
Average CPU utilization

Target value
30

Instances need
30 seconds warm up before including in metric

- I gave **tags value** (optional) as **my-ca2-instance** and click **Next**.

Add tags Info

Add tags to help you search, filter, and track your Auto Scaling group across AWS. You can also choose to automatically add these tags to instances when they are launched.

(i) You can optionally choose to add tags to instances (and their attached EBS volumes) by specifying tags in your launch template. We recommend caution, however, because the tag values for instances from your launch template will be overridden if there are any duplicate keys specified for the Auto Scaling group.

Tags (1)		
Key	Value - optional	Tag new instances
Name	my-ca2-instance	<input checked="" type="checkbox"/>
Add tag		
49 remaining		

[Cancel](#) [Previous](#) **Next**

- It can be seen that **my-ca2-ASG** has **created** perfectly.

Auto Scaling groups (2) Info

Search your Auto Scaling groups

[Create an Auto Scaling group](#)

<input type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
<input type="checkbox"/>	my-ca2-ASG	my-ca2-ASG-template Version Default	3	-	3	2	5	us-east-1a, us-east-1b, us-east-1c, us-ea..

- There are **3 instances** are **running** with the **name of my-ca2-instance** which I gave as a **tag** above while **making my-ca2-ASG**.
- All three has **come again** after **deleting** as a **warm up**.

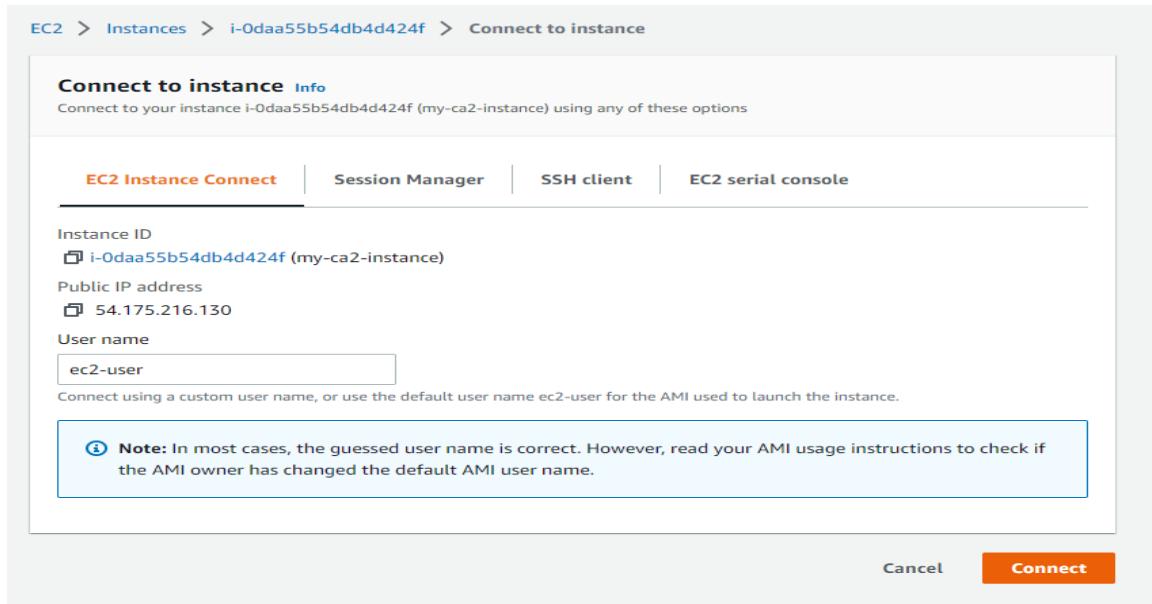
Instances (3/9) Info

Find instance by attribute or tag (case-sensitive)

[Connect](#) [Instance state ▾](#) [Actions ▾](#) [Launch in](#)

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
<input checked="" type="checkbox"/>	my-ca2-instance	i-00734be3f5205a86a	Running	t2.micro	2/2 checks passed	No alarms	+	us-east-1d	ec2-18-234-112-135.co...
<input type="checkbox"/>	web-server-ca-3	i-0361c231631b9dc42	Running	t2.micro	2/2 checks passed	No alarms	+	us-east-1c	ec2-3-86-209-4.comput...
<input type="checkbox"/>	web-server-ca-5	i-0c14dced99e9385a	Running	t2.micro	2/2 checks passed	No alarms	+	us-east-1c	ec2-3-85-162-4.comput...
<input checked="" type="checkbox"/>	my-ca2-instance	i-05d60c10d618238c8	Running	t2.micro	Initializing	No alarms	+	us-east-1c	ec2-18-205-23-99.com...
<input type="checkbox"/>	web-server-ca-4	i-00cb4a61d4631810c	Running	t2.micro	2/2 checks passed	No alarms	+	us-east-1c	ec2-44-211-216-143.co...
<input type="checkbox"/>	web-server-ca-1	i-0520681c6d1410ed7	Running	t2.micro	2/2 checks passed	No alarms	+	us-east-1c	ec2-54-163-184-177.co...
<input type="checkbox"/>	web-server-ca-2	i-03c98c1d50463d656	Running	t2.micro	2/2 checks passed	No alarms	+	us-east-1c	ec2-54-152-191-42.co...
<input type="checkbox"/>	November-28-instance	i-0daf4f226902e4a7a	Running	t2.micro	2/2 checks passed	No alarms	+	us-east-1b	-
<input checked="" type="checkbox"/>	my-ca2-instance	i-01574218fc0673331	Running	t2.micro	2/2 checks passed	No alarms	+	us-east-1b	ec2-52-3-234-123.com...

- Open one instance, Go to Connect inside it.
- Click Connect.



- I used the stress parameters script which has been given in Moodle.
- I applied first command of this script in this Linux instance.

```

Amazon Linux 2 AMI
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-22-85 ~]$ sudo amazon-linux-extras install epel -y
Installing epel-release
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-epel amzn2extra-kernel-5.10
17 metadata files removed
6 sqlite files removed
0 metadata files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2extra-docker
amzn2extra-epel
amzn2extra-kernel-5.10
(1/9): amzn2-core/2/x86_64/group_gz
(2/9): amzn2-core/2/x86_64/updateinfo
(3/9): amzn2extra-epel/2/x86_64/primary_db
(4/9): amzn2extra-kernel-5.10/2/x86_64/updateinfo
(5/9): amzn2extra-epel/2/x86_64/updateinfo
(6/9): amzn2extra-docker/2/x86_64/updateinfo
(7/9): amzn2extra-docker/2/x86_64/primary_db
(8/9): amzn2extra-kernel-5.10/2/x86_64/primary_db
(9/9): amzn2-core/2/x86_64/primary_db
Resolving Dependencies
--> Running transaction check
--> Package epel-release.noarch 0:7-11 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package           Arch      Version
=====
Installing:
  epel-release    noarch   7-11

=====

```

- After that, I applied **second command**.

```
[ec2-user@ip-172-31-22-85 ~]$ sudo yum install stress -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
228 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
--> Package stress.x86_64 0:1.0.4-16.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

Transaction Summary

Package           Arch
Installing:      stress          x86_64

Install 1 Package

Total download size: 39 k
Installed size: 94 k
Downloading packages:
warning: /var/cache/yum/x86_64/2/epel/packages/stress-1.0.4-16.el7.x86_64.rpm: Public key for stress-1.0.4-16.el7.x86_64.rpm is not installed
stress-1.0.4-16.el7.x86_64.rpm
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
Importing GPG key 0x352C64E5:
  Userid : "Fedora EPEL (7) <epel@fedoraproject.org>"
  Fingerprint: 91e9 7d7c 4a5e 96f1 7f3e 888f 6a2f aea2 352c 64e5
  Package : epel-release-7-11.noarch (@amzn2extra-epel)
  From   : /etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : stress-1.0.4-16.el7.x86_64
  Verifying   : stress-1.0.4-16.el7.x86_64

Installed:
```

- At last, I gave the **stress command** with **30-cpu** and **timeout-60**.

```
From      : /etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : stress-1.0.4-16.el7.x86_64
  Verifying   : stress-1.0.4-16.el7.x86_64

Installed:
  stress.x86_64 0:1.0.4-16.el7

Complete!
[ec2-user@ip-172-31-22-85 ~]$
[ec2-user@ip-172-31-22-85 ~]$ sudo stress --cpu 30 --timeout 60
stress: info: [9022] dispatching hogs: 30 cpu, 0 io, 0 vm, 0 hdd
```

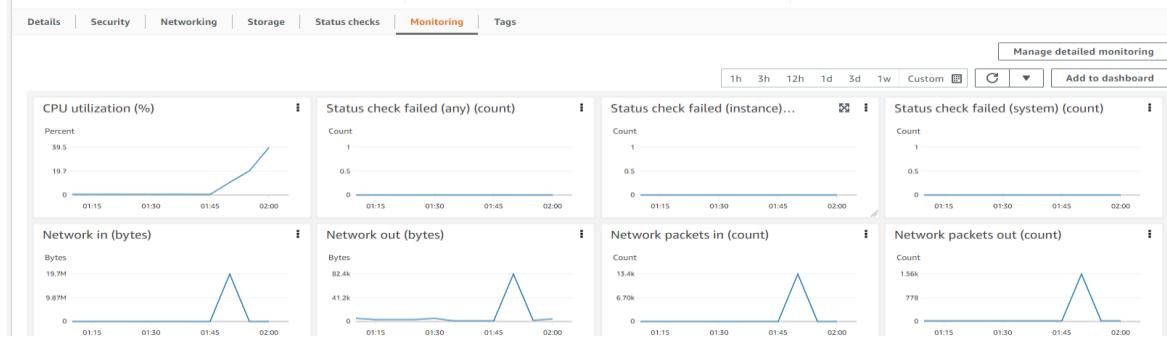
- It wasn't showing me any **stress**.
- So that I applied the **stress command** with **60-cpu** and **timeout-120**.

```
[ec2-user@ip-172-31-22-85 ~]$
[ec2-user@ip-172-31-22-85 ~]$
[ec2-user@ip-172-31-22-85 ~]$
[ec2-user@ip-172-31-22-85 ~]$ sudo stress --cpu 30 --timeout 60
stress: info: [9125] dispatching hogs: 30 cpu, 0 io, 0 vm, 0 hdd

stress: info: [9125] successful run completed in 60s
[ec2-user@ip-172-31-22-85 ~]$
[ec2-user@ip-172-31-22-85 ~]$ sudo stress --cpu 60 --timeout 120

i-0daa55b54db4d424f (my-ca2-instance)
PublicIPs: 54.175.216.130 PrivateIPs: 172.31.22.85
```

- Go to **inside instance** again and **click on Monitoring**.
- You can see the monitoring below:



- It can be seen that I have **two instances** here with the **name my-ca2-instance**.
- **Terminated the one instance** from them.

Instances (1/8) Info								
<input type="button" value="C"/> Connect Instance state ▲ Actions ▼ Launch Instances								
<input type="text"/> Find instance by attribute or tag (case-sensitive)								
	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	
<input type="checkbox"/>	November-28-instance	i-062f03188b2aee3d1	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1a		
<input checked="" type="checkbox"/>	my-ca2-instance	i-0daa55b54db4d424f	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1d		
<input type="checkbox"/>	web-server-ca-5	i-0c14dc6ed99e9385a	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	web-server-ca-4	i-00cb4a61d4631810c	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	web-server-ca-1	i-0520681c6d1410ed7	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	web-server-ca-2	i-03c98c1d50463d656	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	DigiZilla	i-088db5520d78ee881	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	my-ca2-instance	i-0ef2bdb799f46342a	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		

- But it can be seen that the **terminated instance** has **come again**.
- That's mean, **Auto scaling is working properly**.

Instances (9) Info								
<input type="text"/> Find instance by attribute or tag (case-sensitive)								
	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	
<input type="checkbox"/>	November-28-instance	i-062f03188b2aee3d1	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1a		
<input type="checkbox"/>	my-ca2-instance	i-0daa55b54db4d424f	Terminated QQ	t2.micro	-	No alarms +	us-east-1d	
<input type="checkbox"/>	web-server-ca-5	i-0c14dc6ed99e9385a	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	web-server-ca-4	i-00cb4a61d4631810c	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	web-server-ca-1	i-0520681c6d1410ed7	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	web-server-ca-2	i-03c98c1d50463d656	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	DigiZilla	i-088db5520d78ee881	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	my-ca2-instance	i-0ef2bdb799f46342a	Running QQ	t2.micro	2/2 checks passed No alarms	+ us-east-1c		
<input type="checkbox"/>	my-ca2-instance	i-0fd77f6f24d43c870	Running QQ	t2.micro	1 Initializing QQ	No alarms +	us-east-1e	

For Extra Marks

- For Auto scaling group **working** in VPC that I **created** in **Part 2** above as **ca2-2021384-vpc**.
- I did **some modifications** in **ASG template**.
- First, **Go to inside the my-ca2-ASG-template**.
- Click on **Actions** and select **Modify template**.

EC2 > Launch templates > my-ca2-ASG-template

my-ca2-ASG-template (lt-0a67a5b98f8a41357)

Actions ▾ Delete template

Launch template details

Launch template ID: lt-0a67a5b98f8a41357 Launch template name: my-ca2-ASG-template Default version: 1 Owner: arn:aws:sts::066970063753:assumed-role/voclabs/user2189945=2021384@student.cct.ie

Details | Versions | Template tags

Launch template version details

Version: 1 (Default) Description: version1 Date created: 2022-12-03T02:54:32.000Z

Actions ▾ Delete template version

Create new version Modify template (Create new version) Set default version Create Auto Scaling group

- **Template name was same.**
- But I **created** new **security group** with the name **My-ca2-xm-sg**.
- I gave **description** as **ASG-EM-SG**.
- I **selected** my **ca2-2021384-vpc**.

▼ Network settings [Info](#)

Subnet Info

Don't include in launch template [Create new su](#)

When you specify a subnet, a network interface is automatically added to your template.

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Select existing security group [Create security group](#)

Security group name - required

My-ca2-xm-sg

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length: 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-:/()#,@+=;&!\$*

Description - required [Info](#)

ASG-EM-SG

VPC - required [Info](#)

vpc-00f0082cb1b89c16f (ca2-2021384-vpc-vpc)
10.0.0.0/16

Inbound security groups rules

No security group rules are currently included in this template. Add a new rule to include it in the launch tem

Add security group rule

- Then I added two inbound rules first with the type of ssh and other one with HTTP.
- And set their CIDR's as 0.0.0.0/0.
- Click on create.

Inbound security groups rules

Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type Info: ssh Protocol Info: TCP Port range Info: 22

Source type Info: Custom Source Info: Add CIDR, prefix list or security Description - optional Info: e.g. SSH for admin desktop
0.0.0.0/0

Security group rule 2 (TCP, 80, 0.0.0.0/0)

Type Info: HTTP Protocol Info: TCP Port range Info: 80

Source type Info: Custom Source Info: Add CIDR, prefix list or security Description - optional Info: e.g. SSH for admin desktop
0.0.0.0/0

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Add security group rule

- After that go inside the my-ca2-ASG auto scaling group.
- Edit the Launch template.
- Select Version as Latest (2) which I just have set above.

EC2 > Auto Scaling groups > my-ca2-ASG

Edit my-ca2-ASG

Launch template Info Switch to launch configuration

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.
my-ca2-ASG-template

Create a launch template

Version
Latest (2)

Description
Version 1

AMI ID
ami-0b0dcb5067f052a63

Key pair name
Year2-Cloud-key

Launch template
my-ca2-ASG-template
lt-0a67a5b98f8a41357

Security groups
-

Security group IDs
-

Instance type
t2.micro

Request Spot Instances
No

- Go down and **edit** the **networks** option **in this** launch template **edit page**.
- I selected all the subnets of **ca2-2021384-vpc**.

AMI ID: ami-0b0dc5067f052a63
Key pair name: Year2-Cloud-key
Security groups: -
Request Spot Instances: No

Subnets:

- vpc-00f0082cb1b89c16f (ca2-2021384-vpc-vpc)
 - us-east-1a | subnet-0df569a38f0154423 (ca2-2021384-vpc-subnet-private1-us-east-1a) 10.0.128.0/20
 - us-east-1a | subnet-066d48d0f92b4dc95 (ca2-2021384-vpc-subnet-public1-us-east-1a) 10.0.0.0/20
 - us-east-1b | subnet-08c42bcc23c9abbe (ca2-2021384-vpc-subnet-public2-us-east-1b) 10.0.16.0/20
 - us-east-1b | subnet-0824c40921c104522
- Select Availability Zones and subnets
- us-east-1a | subnet-0df569a38f0154423 (ca2-2021384-vpc-subnet-private1-us-east-1a) 10.0.128.0/20
- us-east-1a | subnet-066d48d0f92b4dc95 (ca2-2021384-vpc-subnet-public2-us-east-1b)

- It can be seen that my **new instances with the personal VPC ca2-2021384-vpc are running**.

Instance	ID	Status	Type	Metrics	Alarms	Zone
my-ca2-instance	i-09efb1546eb14d986	Terminated	t2.micro	-	No alarms	us-east-1b
my-ca2-instance	i-016d016bdf1bcb817	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b
my-ca2-instance	i-04da15def1e800ab7	Terminated	t2.micro	-	No alarms	us-east-1e

Instance: i-016d016bdf1bcb817 (my-ca2-instance)

AMI Role: IAM Role

Instance details:

- Platform: Amazon Linux (Inferred)
- AMI ID: ami-0b0dc5067f052a63
- Monitoring: disabled

- These have also come back after terminated.
- That's mean Auto scaling is working properly.

Instances (11) Info										
Find instance by attribute or tag (case-sensitive) C Connect Instance state Actions Launch										
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv6		
my-ca2-instance	i-0441b66f5b24825f4	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1a	-	-		
web-server-ca-5	i-0c14dc6ed99e9385a	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-3-94-8-66.compute...	3.94.8.66		
web-server-ca-4	i-00cb461d4631810c	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-3-82-114-133.com...	3.82.114.1		
web-server-ca-1	i-0520681c6d1410ed7	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-3-83-240-24.comp...	3.83.240.2		
web-server-ca-2	i-03c98c1d5046d656	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-18-204-212-220.co...	18.204.212		
DigiZilla	i-088db5520d78ee881	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1c	ec2-44-204-27-6.comp...	44.204.27.		
my-ca2-instance	i-02cf6647ce2830f18	Terminated	t2.micro	-	No alarms	+ us-east-1c	-	-		
November-28-instance	i-099489ab710022bb0	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1b	-	-		
my-ca2-instance	i-09efb1546e6f4d986	Terminated	t2.micro	-	No alarms	+ us-east-1b	-	-		
my-ca2-instance	i-016d016bdf1bcb817	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1b	-	-		
my-ca2-instance	i-04da15def1e800ab7	Terminated	t2.micro	-	No alarms	+ us-east-1e	-	-		

Part 6: Amazon Linux VM instance

- In **EC2**, navigate **instances** and **click on launch instance**.
- I have created a new **Amazon Linux instance**.
- I gave it to name **DigiZilla**.
- I made it by **following the same procedure** which I used in **part 4** above for making 5 instances.



Name and tags [Info](#)

Name Add additional tag

▼ 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

Recent | Quick Start

Amazon Linux  macOS  Ubuntu  Windows  Red Hat  S > 

Browse more AMIs Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

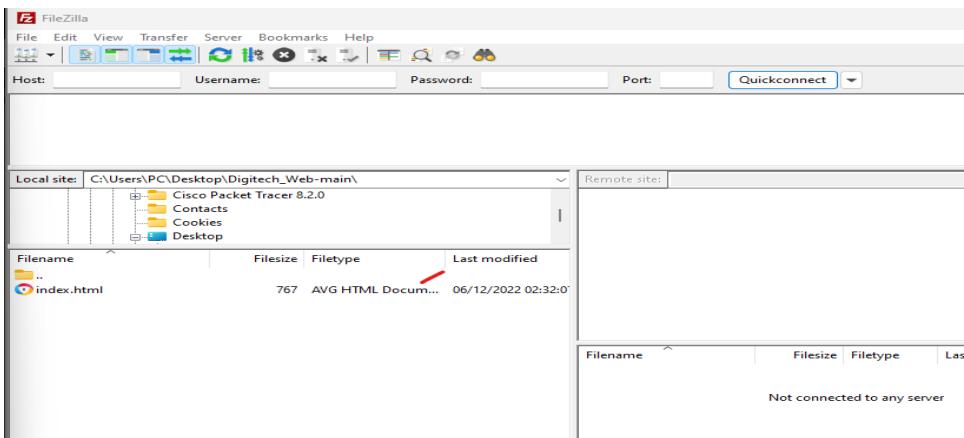
Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type
ami-0b0dcba067f052a63 (64-bit (x86)) / ami-01b5ec3ed8678d8b7 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

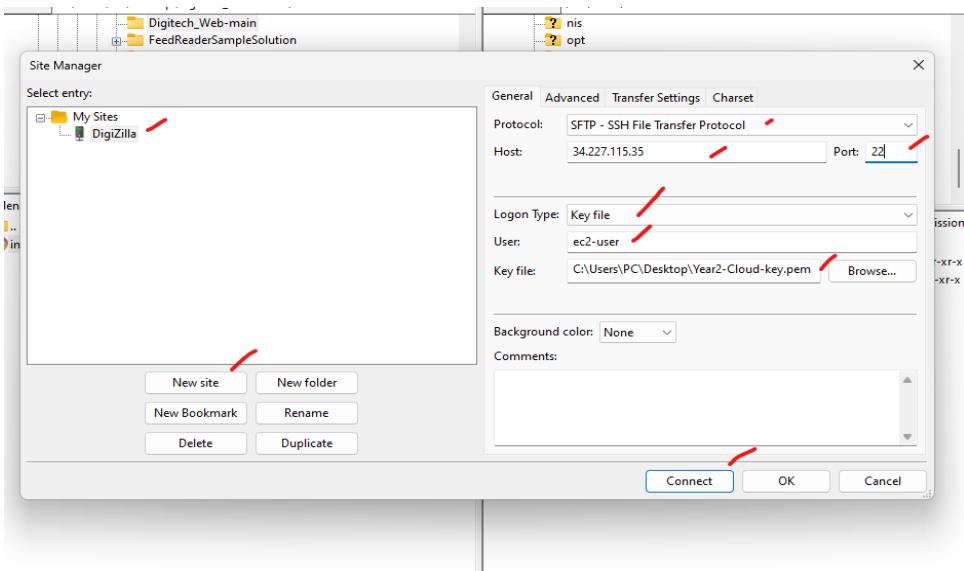
- **Amazon Linux DigiZilla has created.**

<input type="checkbox"/> web-server-ca-2	i-03c98c1d50463d656	 Running  t2.micro	 2/2 checks passed No alarms + us-east-1c	ec2-54-89-250-132.co...	54.89.250.132
<input type="checkbox"/> DigiZilla 	i-088db5520d78ee881	 Running  t2.micro	 2/2 checks passed No alarms + us-east-1c	ec2-52-87-229-194.co...	52.87.229.194
<input type="checkbox"/> my-ca2-instance	i-0cb083ea537deef15	 Running  t2.micro	 2/2 checks passed No alarms + us-east-1f	ec2-3-239-95-50.comp...	3.239.95.50

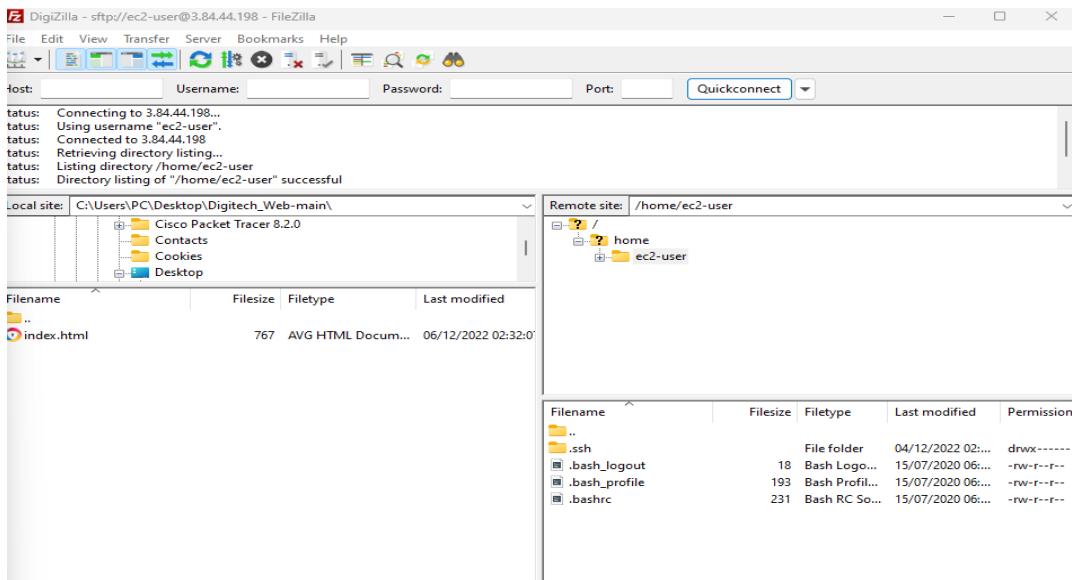
- I selected the **Digitech** website **inside** from my **device** in **FileZilla**.



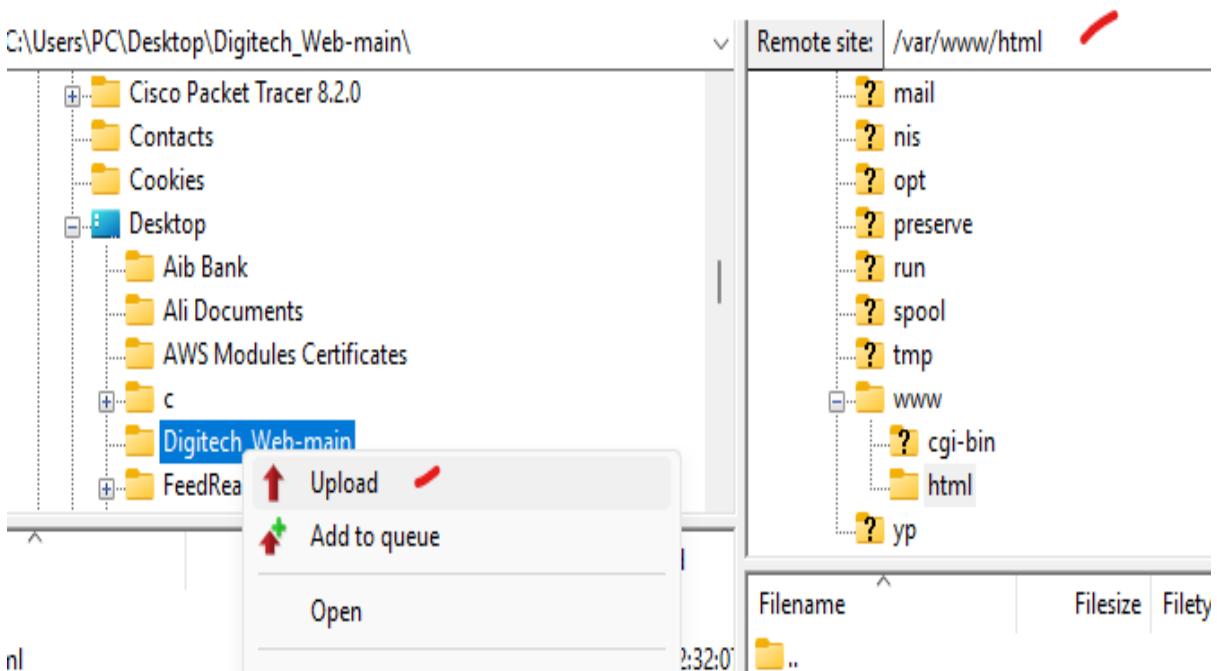
- In top left of FileZilla software, Click on **File**.
- Go to **Site manager** and Click on **new site**.
- I gave **name as DigiZilla**.
- Select **secure ftp** in **Protocol** drop down menu.
- In **Host**, paste the **IP address** of **DigiZilla instance**, which I made in AWS cloud.
- In **Logon type**, select **Key file**.
- I wrote **ec2-user**.
- Browse and gave the device **location** of my **Year2-cloud-key** and Click **Connect**.



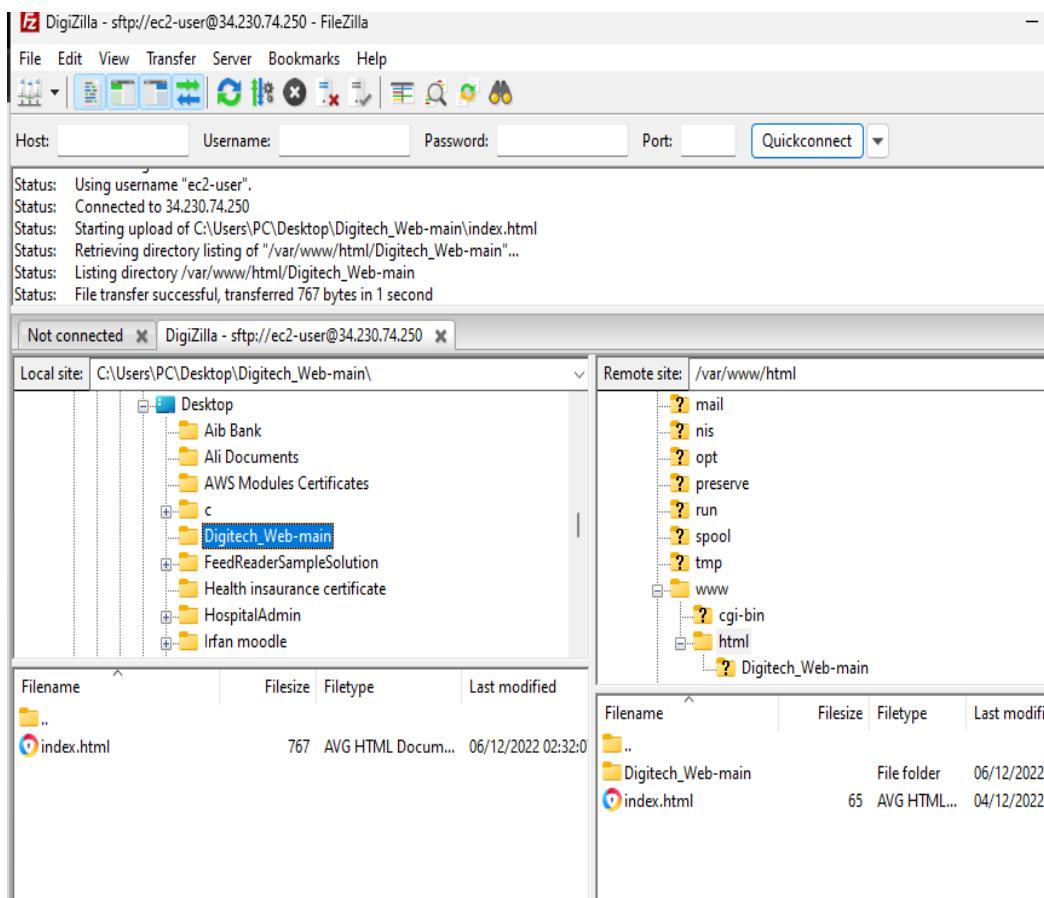
- It can be seen that, DigiZilla is successfully connected.



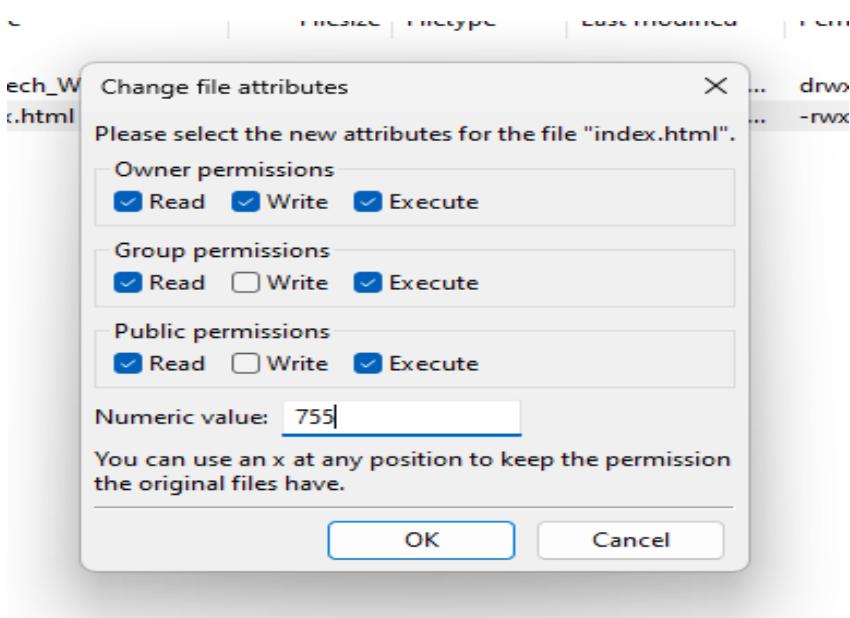
- In **Remote site** section, **expand the root**.
- Then **expand the folder var**, inside var **expand www**.
- Inside www, **expand html folder**.
- Right click on Digitech web and click on Upload.**



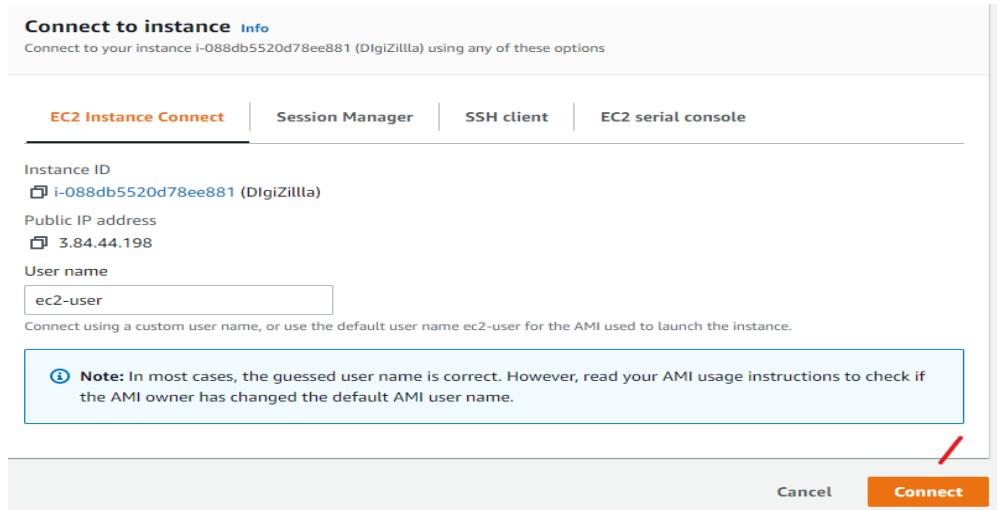
- It can be seen in screenshot below that **All files are transferred successful**.



- Right click on **index.html** file in **remote site section** and click on **File permission**.
- Make sure the **Numerical value** is **755**.



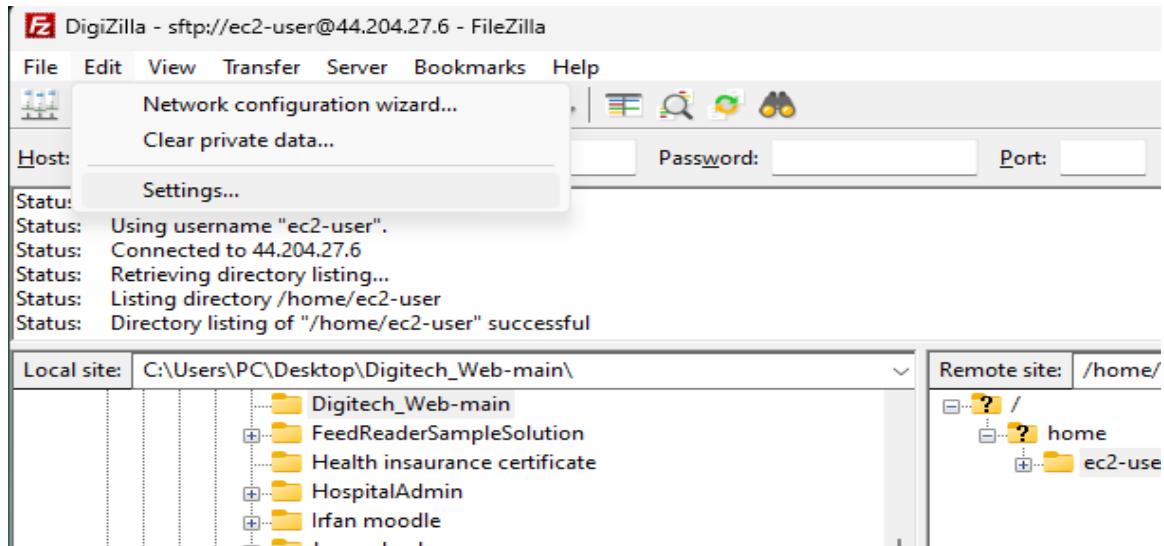
- Go inside the **DigiZilla** instance.
- Click **Connect**.



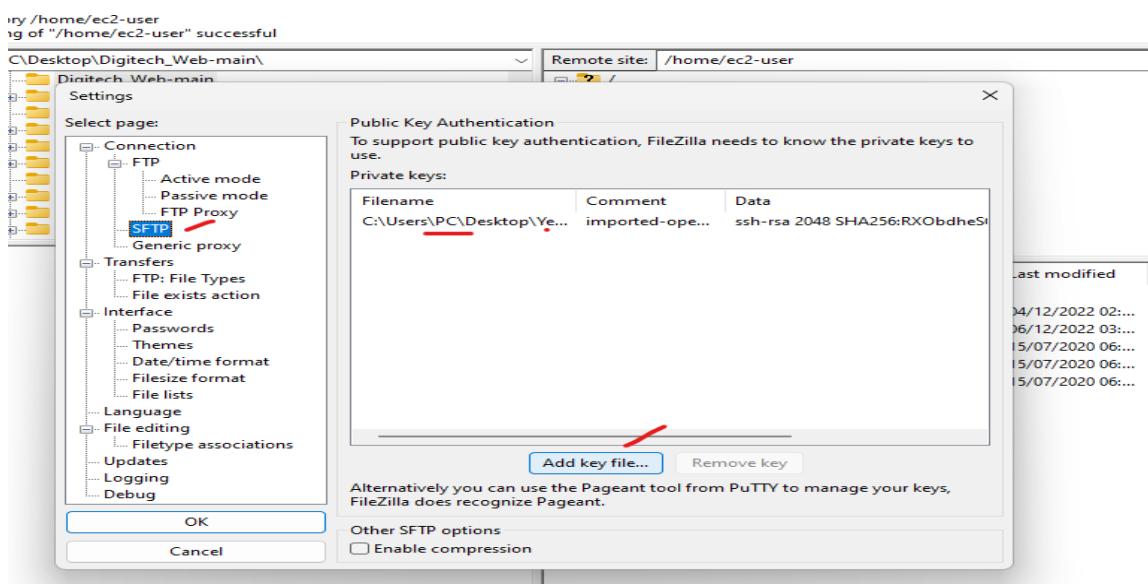
- I gave **Web permissions** which has been given on Moodle.
- **First**, I gave command as **sudo chown -R ec2-user:ec2-user /var/www/html**.
- **Second** command as **sudo chmod -R 755 /var/www/html**.

```
Last login: Sun Dec 4 02:40:09 2022 from ec2-18-206-107-29.compute-1.amazonaws.com
[ec2-user@ip-172-31-92-78 ~]$ sudo chown -R ec2-user:ec2-user /var/www/html
[ec2-user@ip-172-31-92-78 ~]$ sudo chmod -R 755 /var/www/html
```

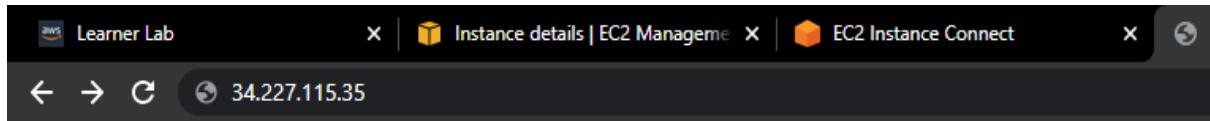
- I copied the Public IP address and paste into Google and searched it.
- But it showed me only bootstrap script while searching on google.
- Then, went to Edit on FileZilla and navigate Settings.



- Click SFTP option on left menu.
- Click on Add key file option.
- Navigate the same key pair from device which I used for making the DigiZilla instance.
- And added it.



- I uploaded the **Digitech** web **again** and **followed** the **whole procedure again**.
- And it **can be seen** in the screenshot below that the **Digitech** web is **working successfully**.



Digitech Company



Webmaster : Muhammad Sajjad Haider

Student Number : 2021384

Challenge Task 2:

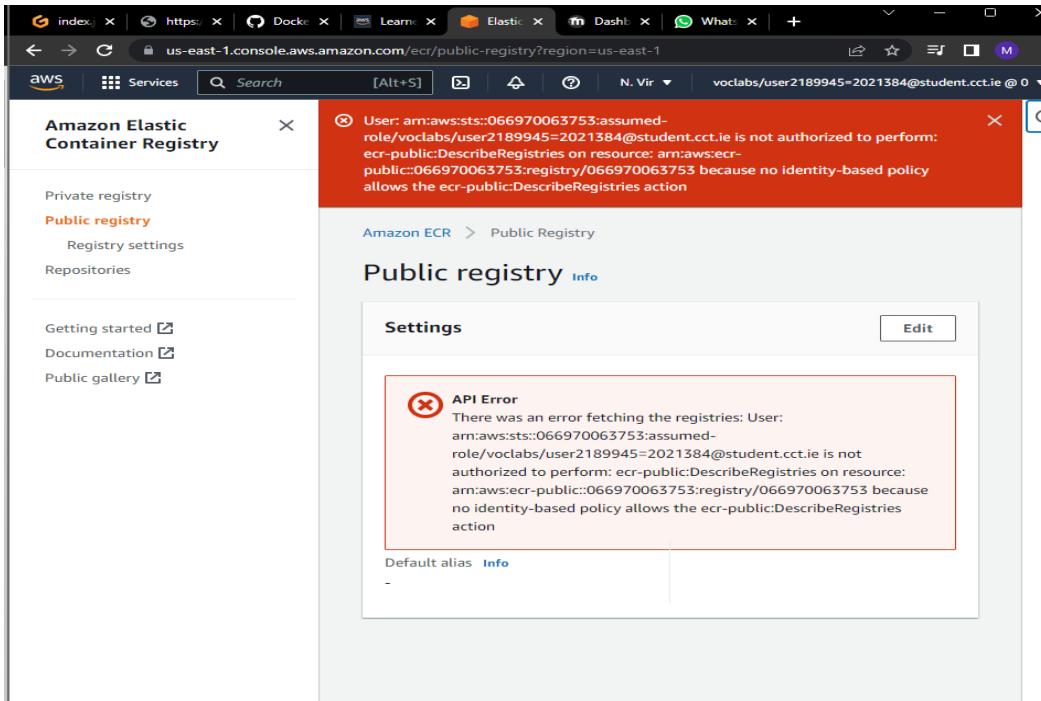
- I typed **ecs** for **Elastic container registry** in **search bar of services**.
- Click on **Repositories**.
- Click on **create repository**.

The screenshot shows the AWS ECR (Amazon Elastic Container Registry) interface. On the left, there's a sidebar with links for 'Private registry', 'Public registry', and 'Repositories'. The main area is titled 'Amazon ECR > Repositories' and shows a 'Private repositories' section. At the top of this section is a 'Create repository' button, which is highlighted with a red arrow. Below it is a search bar labeled 'Find repositories'. A table header is visible, showing columns for 'Repository name', 'URI', 'Created at', 'Tag immutability', and 'Scan frequency'. A message at the bottom states 'No repositories' and 'No repositories were found'.

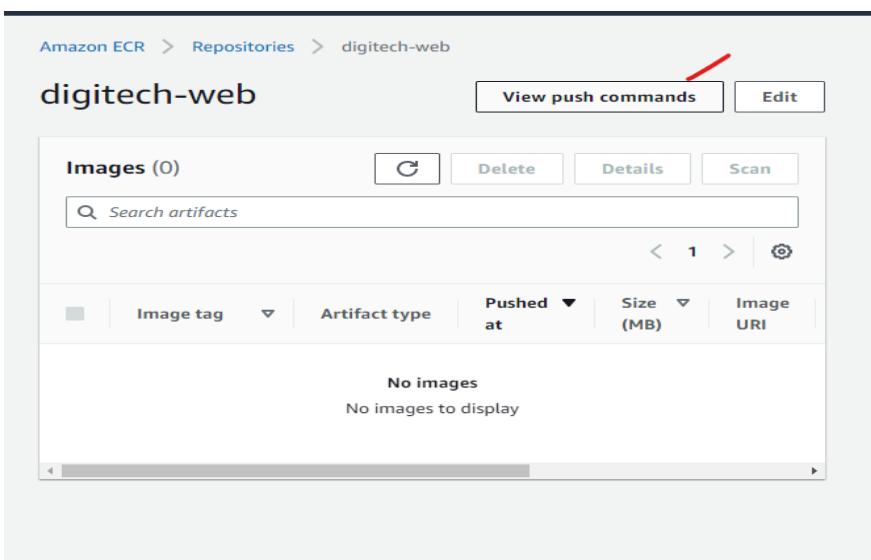
- I made it **Public**.
- I gave **repository name** as **digitech-web**.

This screenshot shows the 'Create repository' wizard. In the 'General settings' step, the 'Visibility settings' section is open, showing the 'Info' link and a note about changing visibility. Two options are available: 'Private' (radio button unselected) and 'Public' (radio button selected, highlighted with a red arrow). A note below states: 'Once a repository is created, the visibility setting of the repository can't be changed.' In the 'Detail' step, the 'Repository name' field contains 'public.ecr.aws/registry-alias/digitech-web' (with the 'digitech-web' part highlighted with a red arrow), and a note indicates the name must start with a letter and contain lowercase letters, numbers, hyphens, underscores, periods, and forward slashes. A note at the bottom of the step states: 'A default alias is associated with your public registry once your first public repository is created. The registry alias is displayed as a prefix to the repository name in the repository URI. A custom alias can be requested on the Registry settings page.'

- I tried many times but it gave me error while making it.
- According to it I am not authorized to perform it as Public on a student account.



- So, I made it as private with the same name as digitech-web.
- It was created successfully.
- Then went inside the digitech-web.
- Click on View Push Commands.



- I needed to copy all these commands and run into my online visual studio where I created my Docker file.
- I tried very hard and install AWS tools also in my online visual studio.
- But it didn't work.
- It again gave me error as no access.

Make sure that you have the latest version of the AWS Tools for PowerShell and Docker installed. For more information, see [Getting Started with Amazon ECR](#).

Use the following steps to authenticate and push an image to your repository. For additional registry authentication methods, including the Amazon ECR credential helper, see [Registry Authentication](#).

1. Retrieve an authentication token and authenticate your Docker client to your registry.

Use AWS Tools for PowerShell:

```
powershell (Get-ECRLoginCommand).Password | docker login --username AWS --password-stdin 066970063753.dkr.ecr.us-east-1.amazonaws.com
```

2. Build your Docker image using the following command. For information on building a Docker file from scratch see the instructions [here](#). You can skip this step if your image is already built:

```
powershell docker build -t digitech-web .
```

3. After the build completes, tag your image so you can push the image to this repository:

```
powershell docker tag digitech-web:latest 066970063753.dkr.ecr.us-east-1.amazonaws.com/digitech-web:latest
```

4. Run the following command to push this image to your newly created AWS repository:

```
powershell docker push 066970063753.dkr.ecr.us-east-1.amazonaws.com/digitech-web:latest
```

Click

- Here is my online visual studio where I created my docker file.
- This web app is running on port 6565 successfully.

```
version: '3.8'
services:
  app:
    image: node:14-alpine
    working_dir: /app
    volumes:
      - .:/app
    ports:
      - 6565:5000
```

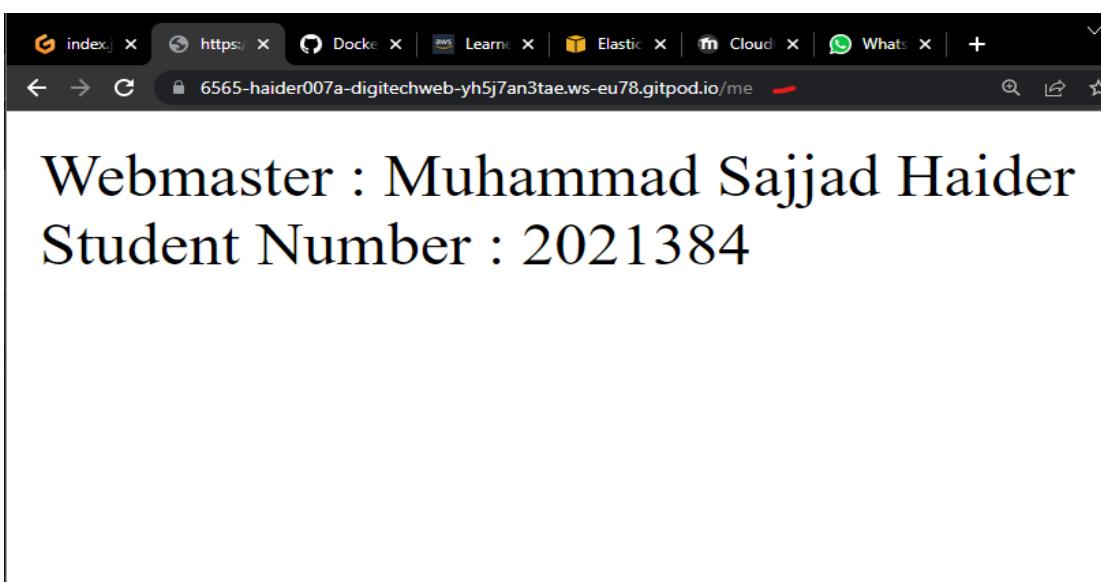
```
const express = require("express");
const app = express();
app.get("/", (req, res) => {
  res.send("Welcome to Digitech Company");
});
app.get("/me", (req, res) => {
  res.send(`Webmaster : Muhammad Sajjad Haider      Student Number : ${req.query.id}`);
});
app.listen(5000, () => {
  console.log("listening");
});
```

Port	State	Action
6565	open (public)	

- It can be seen that Docker file is also running properly in web browser.



- If I type /me next to link, it shows me the name and student number.
- I gave that shots as proof that I didn't have access that's why I couldn't complete.
- Anyhow, now I learnt it properly and know how to work on it.

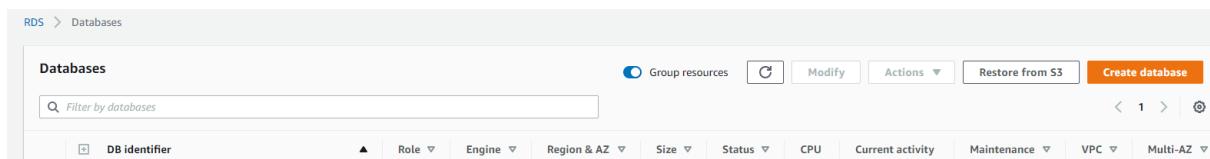


Reference

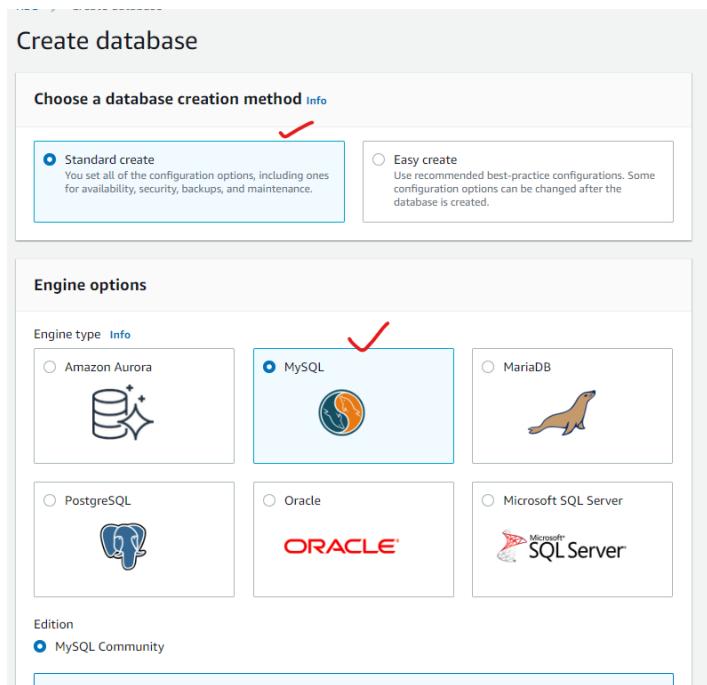
www.youtube.com. (n.d.). *How to Deploy a Docker App to AWS ECS*. [online] Available at: <https://www.youtube.com/watch?v=YDNSItBN15w> [Accessed 11 Dec. 2022].

Challenge Task 3:

- Navigate RDS in **services** and select **Databases** from left menu.
- Click on **Create database**.



- Stick with **Standard create**.
- In **Engine options**, select **MySQL**.



- In **Templates**, select **Free tier**.

Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

Engine Version: MySQL 8.0.28

Templates

Choose a sample template to meet your use case.

- Production: Use defaults for high availability and fast, consistent performance.
- Dev/Test: This instance is intended for development use outside of a production environment.
- Free tier: Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS. [Info](#)

Availability and durability

Deployment options: [Info](#)
 The deployment options below are limited to those supported by the engine you selected above.

- In **Settings**, I gave **ca2-db-2021384** as **identifier**.
- I wrote **admin** in **Master username** and **root1234** in **Master password**.

Settings

DB instance identifier: [Info](#)
 Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in Region.
 ✓

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ **Credentials Settings**

Master username: [Info](#)
 Type a login ID for the master user of your DB instance.
 ✓

1 to 16 alphanumeric characters. First character must be a letter.

Auto generate a password
 Amazon RDS can generate a password for you, or you can specify your own password.

Master password: [Info](#) ✓

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote), @ (at sign).

Confirm master password: [Info](#)

- I selected **db.t2.micro** in **DB instance class**.

Instance configuration

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

DB instance class: [Info](#)

- Standard classes (includes m classes)
- Memory optimized classes (includes r and x classes)
- Burstable classes (includes t classes)

✓
 1 vCPUs 1 GiB RAM Not EBS Optimized

Include previous generation classes

- In **Connectivity**, **Public access** should be **Yes**.
- I selected **default VPC** security group.
- Click **create**.

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 selected.

default ▾

Public access [Info](#)

Yes -
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

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

Existing VPC security groups
Choose one or more options ▾

default X -

Availability Zone [Info](#)

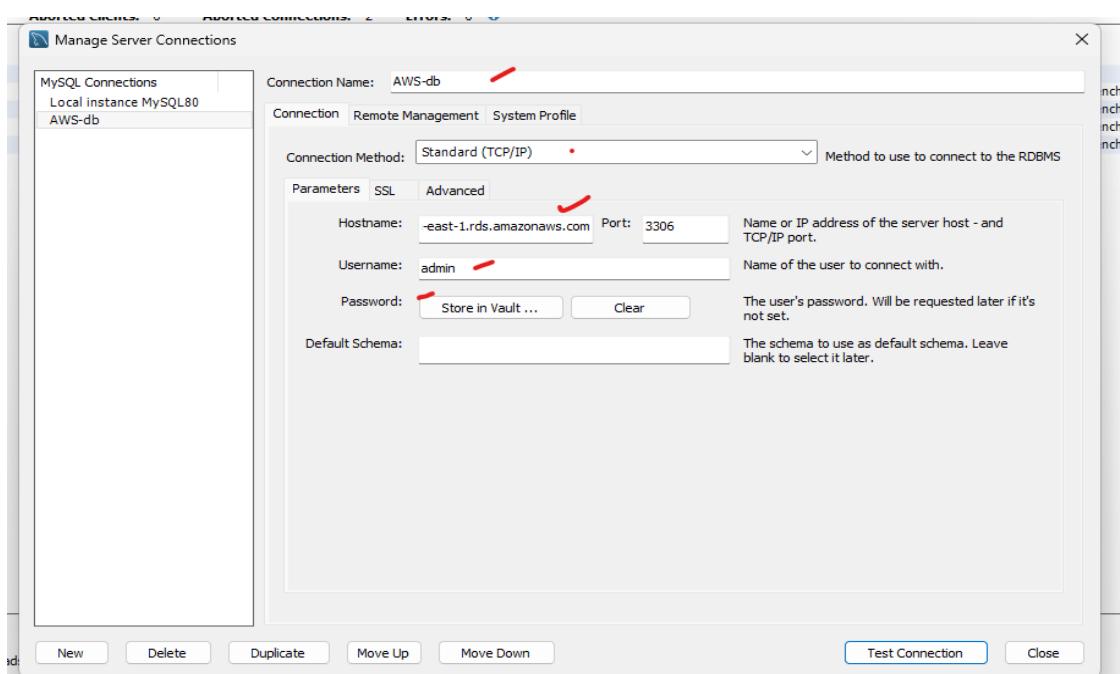
- It took time in **creating** and **back-up**, finally, **created** successfully.

Databases										
<input checked="" type="radio"/> Group resources C Actions Restore from S3 Create database										
<input type="text"/> Filter by databases										
DB identifier	Role	Engine	Region & AZ	Size	Status	CPU	Current activity	Maintenance		
ca2-db-2021384	Instance	MySQL Community	us-east-1d	db.t2.micro	Available	4.00%	0 Connections	none		

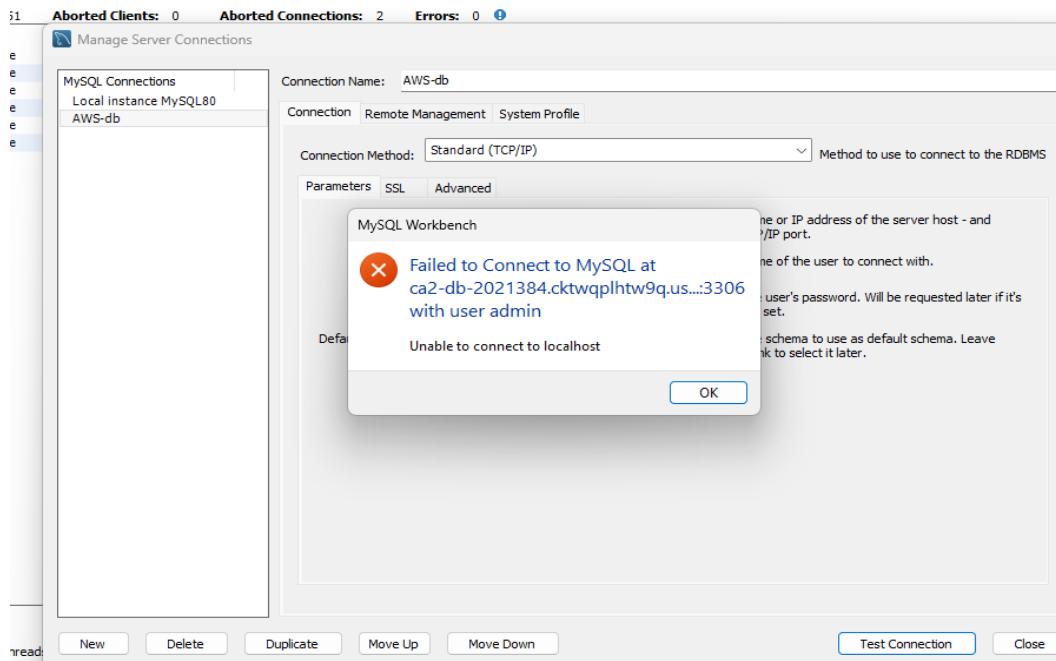
- Open the ca2-db-2021384.
- Copy the highlighted Endpoint.

Connectivity & security		
Endpoint & port	Networking	Security
Endpoint ca2-db-2021384.cktwqlhtw9q.us-east-1.rds.amazonaws.com Port 3306	Availability Zone us-east-1d VPC vpc-0609b8b9b73d5ea21 Subnet group default-vpc-0609b8b9b73d5ea21 Subnets subnet-Oad3e2411e00dd99b subnet-03c616919da748132	VPC security groups default (sg-0a492651ed64c75cd) Active Public accessibility Yes Certificate authority rds-ca-2019 Certificate authority date August 22, 2024, 18:08 (UTC+01:00)

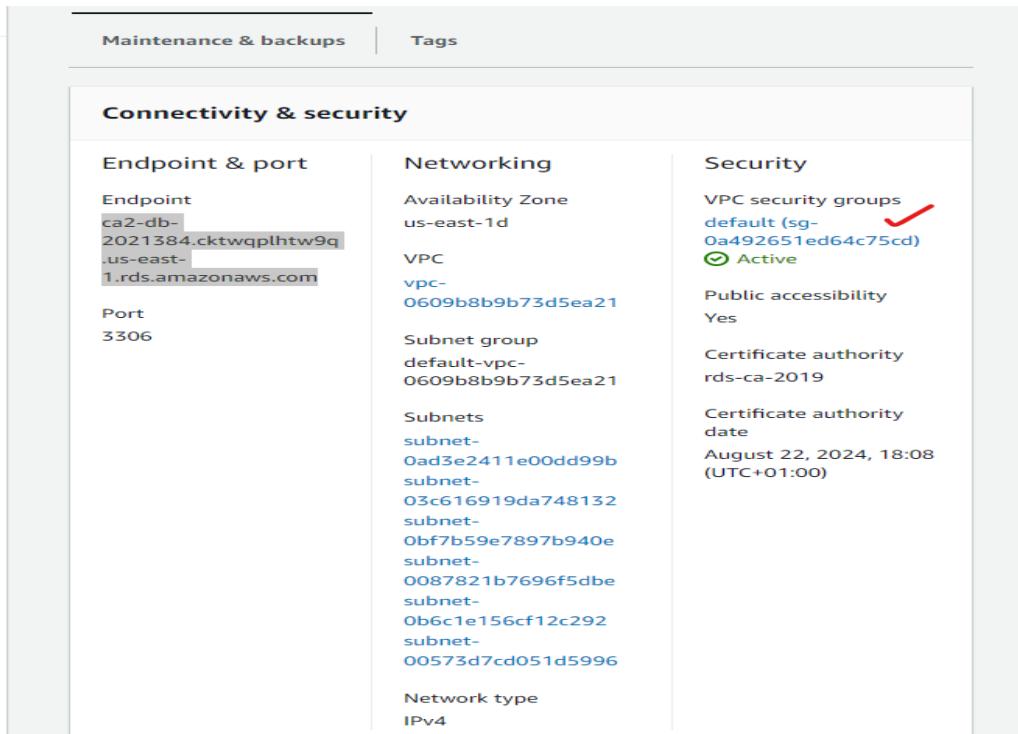
- I used **MySQL workbench** for **connecting** the database **ca2-db-2021384**.
- I gave **connection name** as **AWS-db** while **configuring** the **connection** in **MySQL**.
- Paste** the copied **Endpoint** in **Hostname**.
- Wrote username **admin** and gave **password** which I have set **while making ca2-db-2021384** above.
- Test connection.**



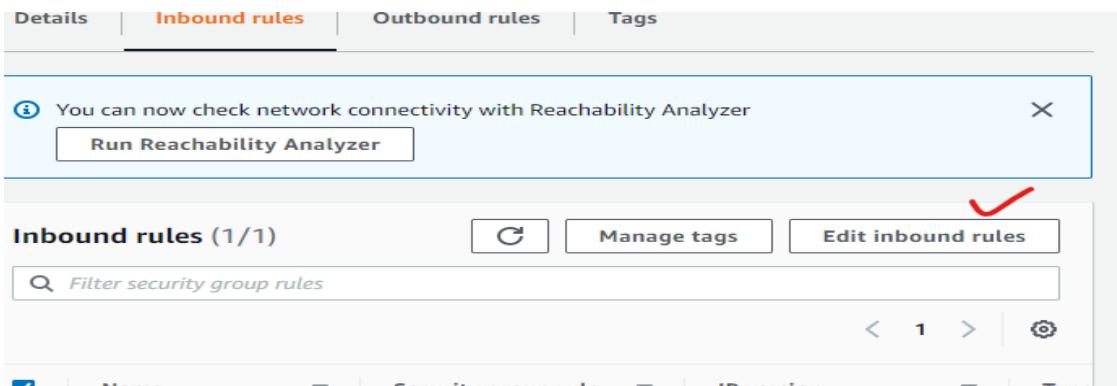
- It can be seen that connection is failed.



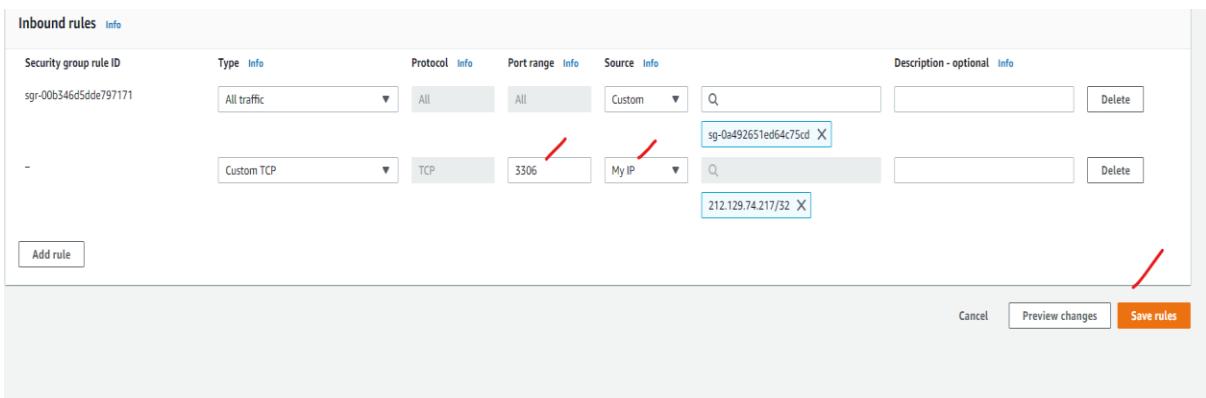
- For troubleshooting, go to default VPC security group in Security.



- Click **edit inbound rules**.



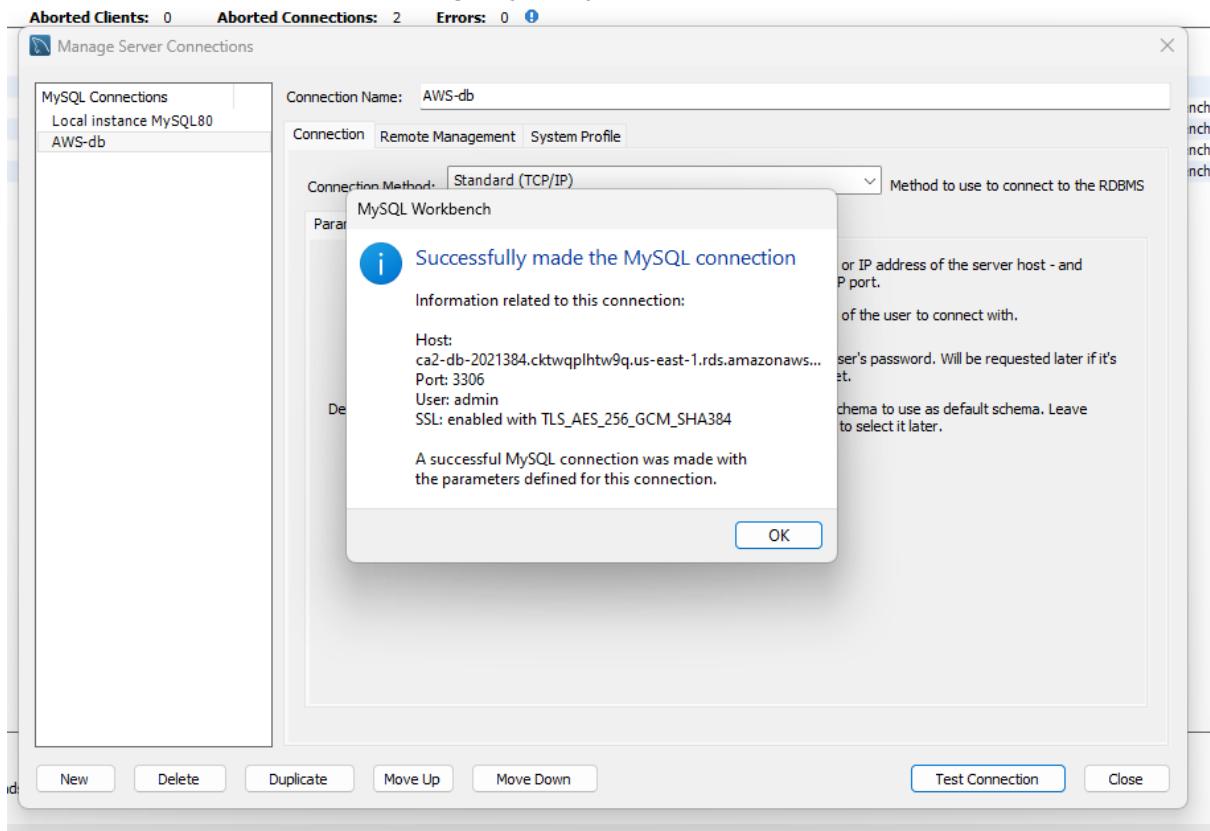
- Add rule.
- Gave the **port range 3306**.
- Set **Source to MyIP**.
- Click **Save rule**.



- **MYSQL/Aurora rule has updated.**

IP version	Type	Protocol	Port range	Source	Description
IPv4	MYSQL/Aurora	TCP	3306	212.129.74.217/32	-
-	All traffic	All	All	sg-0a492651ed64c75c...	-

- **Tested connection again.**
- It can be seen that the **MySQL connection has successfully made.**

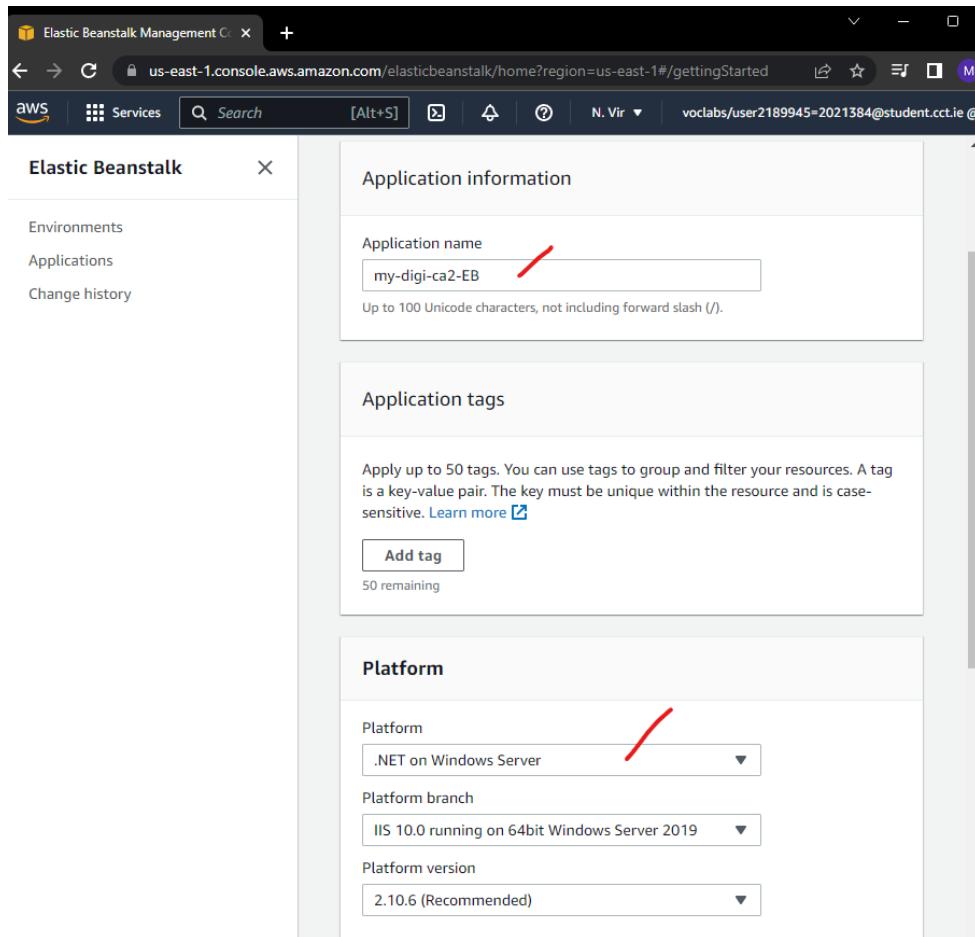


Reference

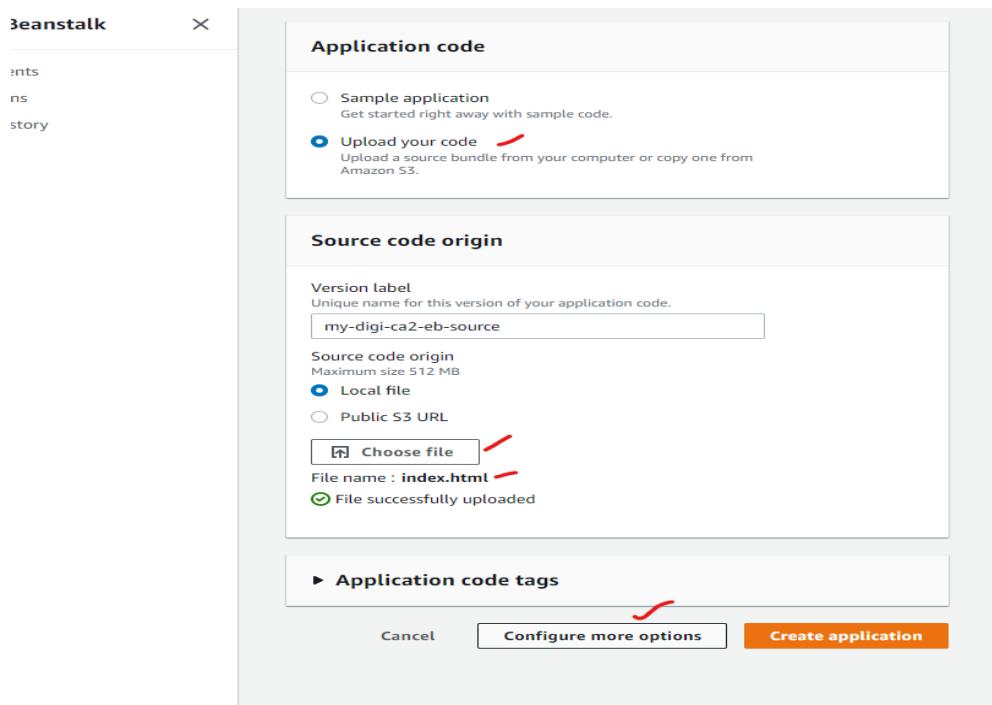
www.youtube.com. (n.d.). *Create a MySQL database on AWS!* [online] Available at: <https://www.youtube.com/watch?v=v3jH1YxJqaY> [Accessed 11 Dec. 2022].

Challenge Task 4:

- Firstly, **Go to compute.**
- **Open Elastic beanstalk** and **click on create application.**
- I gave **name of application as my-digi-ca2-EB.**
- I chose **platform as .NET.**



- In **application code**, **click on your code.**
- In **source code origin** click on **Local file.**
- Select **choose file option** and **browse** the **digitech web page.**
- **Index.html** page has **added.**
- Click on **Configure more options.**



- Stick with **Single instance (Free tier)** in Presets.
- Click on **create app**.

Elastic Beanstalk > Getting started

Configure Mydigica2eb-env

Presets

Start from a preset that matches your use case or choose *Custom configuration* to unset recommended values and use the service's default values.

Configuration presets

- Single instance (*Free Tier eligible*)
- Single instance (using Spot instance)
- High availability
- High availability (using Spot and On-Demand instances)
- Custom configuration

Platform

- It can be seen that my-digi-ca2-EB has created.

All applications							Actions ▾	
<input type="text"/> Filter results matching the display values								
Application name	▲	Environments	▼	Date created	▼	Last modified	▼	ARN
my-digi-ca2-EB				2022-12-10 21:51:06 UTC+0000		2022-12-10 21:51:06 UTC+0000		arn:aws:elasticbeanstalk:us-east-1:0669700637

- Go Inside this application **my-digi-ca2-EB**.
- Click on **Create one now** for creating the Environment.

Application 'my-digi-ca2-EB' environments												Create a new			
<input type="text"/> Filter results matching the display values															
Environment name	▲	Health	▼	Date created	▼	Last modified	▼	URL	▼	Running versions	▼	Platform	▼	Platform state	▼
No environments currently exist for this application.															
Create one now															

- In **Select environment tier** section, Stick with **Web Server Environment**.
- Click on **Select**.

Elastic Beanstalk > Applications > my-digi-ca2-EB

Select environment tier

Amazon Elastic Beanstalk has two types of environment tiers to support different types of web applications. Web servers are standard applications that listen for and then process HTTP requests, typically over port 80. Workers are specialized applications that have a background processing task that listens for messages on an Amazon SQS queue. Worker applications post those messages to your application by using HTTP.

Web server environment
Run a website, web application, or web API that serves HTTP requests.
[Learn more](#)

Worker environment
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule.
[Learn more](#)

[Cancel](#) **Select**

- I gave **Environment name** as **Mydigica2eb-Live**.

- **Domain as Mydigica2eb.**
- Click on **Configure more options.**

Elastic Beanstalk > Applications > my-digi-ca2-EB

Environment information

Choose the name, subdomain, and description for your environment. These cannot be changed later.

Application name
my-digi-ca2-EB

Environment name
Mydigica2eb-Live

Domain
Mydigica2eb.us-east-1.elasticbeanstalk.com

Check availability
Mydigica2eb.us-east-1.elasticbeanstalk.com is available.

Description
Live Environment for Digitech web Application

Platform

Managed platform Platforms published and maintained by AWS

Custom platform Platforms created and managed by you

- I followed the whole procedure again which I did while making application above.

Elastic Beanstalk > Applications > my-digi-ca2-EB

Configure Mydigica2eb-Live

Presets

Start from a preset that matches your use case or choose *Custom configuration* to unset recommended values and use the service's default values.

Configuration presets

Single instance (*Free Tier eligible*)

Single instance (using Spot instances)

High availability

High availability (using Spot and On-Demand instances)

Custom configuration

Platform

IIS 10.0 running on 64bit Windows Server 2019/2.10.6

[Change platform version](#)

Platform

.NET on Windows Server

Platform branch

IIS 10.0 running on 64bit Windows Server 2019

Platform version

2.10.6 (Recommended)

Application code

- Sample application
Get started right away with sample code.
- Existing version
Application versions that you have uploaded for my-digi-ca2-EB.
--- Choose a version ---
- Upload your code
Upload a source bundle from your computer or copy one from Amazon S3.

Version label
Unique name for this version of your application code.

Source code origin
Maximum size 512 MB

- Local file
- Public S3 URL

File name : **Dgitech_Web-main.zip**

File successfully uploaded

► Application code tags

- Click on create Environment.

Database

Engine:	Instance class:	Multi-AZ:
--	--	--
Storage (GB): --		

Edit

Tags

Edit

Tags:
none

Cancel **Previous** **Create environment**

- I tried many times but it gave me error while making it.
- According to it I am not authorized to perform it as Public on a student account.
- I gave that shots as proof that I didn't have access that's why I couldn't complete.
- Anyhow, now I learnt it properly and know how to work on it.

Reference

www.youtube.com. (n.d.). *How To Create And Launch An Application With AWS Elastic Beanstalk*. [online] Available at:
<https://www.youtube.com/watch?v=Mh9Qx-K4UEo> [Accessed 11 Dec. 2022].