

AWS EC2

Demo Document 1

edureka!

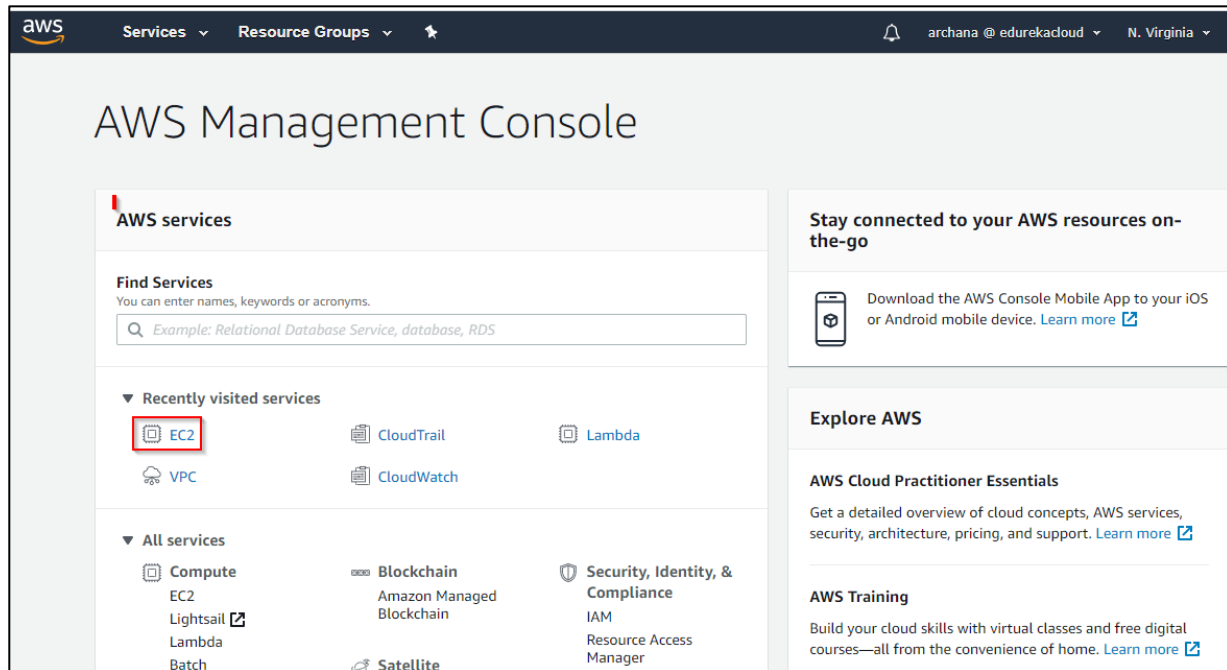
edureka!

© Brain4ce Education Solutions Pvt. Ltd.

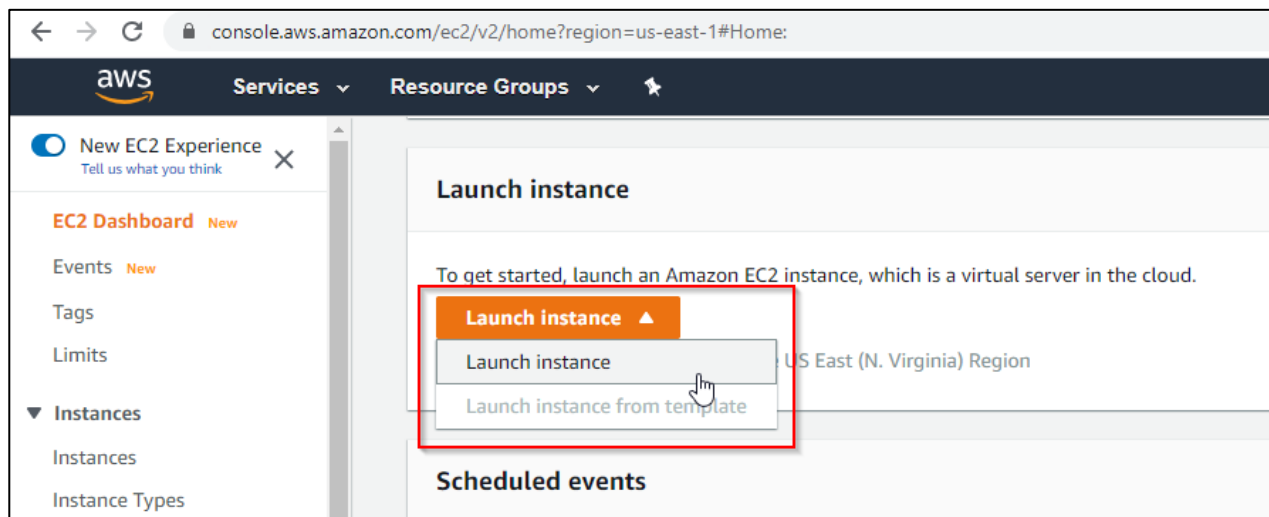
Host your Website Inside your EC2 Instance

Steps to launch an EC2 Instance:

Step 1: In Services menu select EC2 service. Remember EC2 Instance is region specific and EC2 instances created in one region will not be available to other regions.



Step 2: In EC2 dashboard scroll down and click on “Launch”.



Step 3: Select an operating system. For this demo let's select Amazon Linux.

Step 1: Choose an Amazon Machine Image (AMI)

SUSE Linux Enterprise Server 15 SP2 (HVM), SSD Volume Type - ami-03f4c416f489586a3 (64-bit x86) / ami-0d24f1c1ba96d2803 (64-bit Arm) **Select**

Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-0010d386b82bc06f0 (64-bit x86) / ami-00ea3ade739217b29 (64-bit Arm) **Select**

Are you launching a database instance? Try Amazon RDS. [Launch a database using RDS](#)

Microsoft Windows Server 2019 Base - ami-0239d3998515e9ed1 **Select**

Step 4: Select t2.micro (Free tier eligible) and Click on “Next Configure Instance Details”.

Choose an Instance Type

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes

Step 5: Keep all the fields as it is. Click on “Add storage”.

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, instance, and more.

Number of instances 1 [Launch into Auto Scaling Group](#)

Purchasing option ☐ Request Spot instances

Network vpc-eb74a982 (default) [Create new VPC](#)

Subnet No preference (default subnet in any Availability Zone) [Create new subnet](#)

Auto-assign Public IP Use subnet setting (Enable)

Placement group ☐ Add instance to placement group.

Capacity Reservation Open [Create new Capacity Reservation](#)

Step 6: Do not change any thing and click on “Next: Add Tags”.

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/xvda	snap-00525f7708ba0b6dc	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

Step 7: Click on “Add another Tag” and add key and key values, click on “Next Configure Security Group”. (Add tags is an optional key-value pair field, mostly used by user to categorize servers. E.g. Production, UAT, SIT etc. This helps administrator during billing of your AWS account.)

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (127 characters maximum)	Value (255 characters maximum)	Instances	Volumes
Name	Edureka-training	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

Step 8: Choose the security group and make changes as shown below.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach you allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group
☐ Select an existing security group

Security group name: launch-wizard-80

Description: launch-wizard-80 created 2018-12-12T13:51:17.648+05:30

Type	Protocol	Port Range	Source
SSH	TCP	22	Anywhere 0.0.0.0/0, ::/0
HTTP	TCP	80	Anywhere 0.0.0.0/0, ::/0

[Add Rule](#)

Step 9: Click on “Launch”.

Step 7: Review Instance Launch
Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

⚠ Improve your instances' security. Your security group, launch-wizard-80, is open to the world.
Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only. You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

▼ **AMI Details**

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-02e680c4540db351e

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 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.

Root Device Type: ebs Virtualization type: hvm

[Edit AMI](#)

▼ **Instance Type**

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

[Edit instance type](#)

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

Step 10: In “Select an existing key pair or create a new key pair” popup, select “Create a new key pair”. Give a name for your key pair.

Do not forget to download key pair. Otherwise you will not be able to connect to your EC2 instances through SSH terminal (Putty, FileZilla, etc.). Finally click on **Launch Instances**.

Select an existing key pair or create a new key pair ✕

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. [Learn more about removing existing key pairs from a public AMI.](#)

Create a new key pair ▼

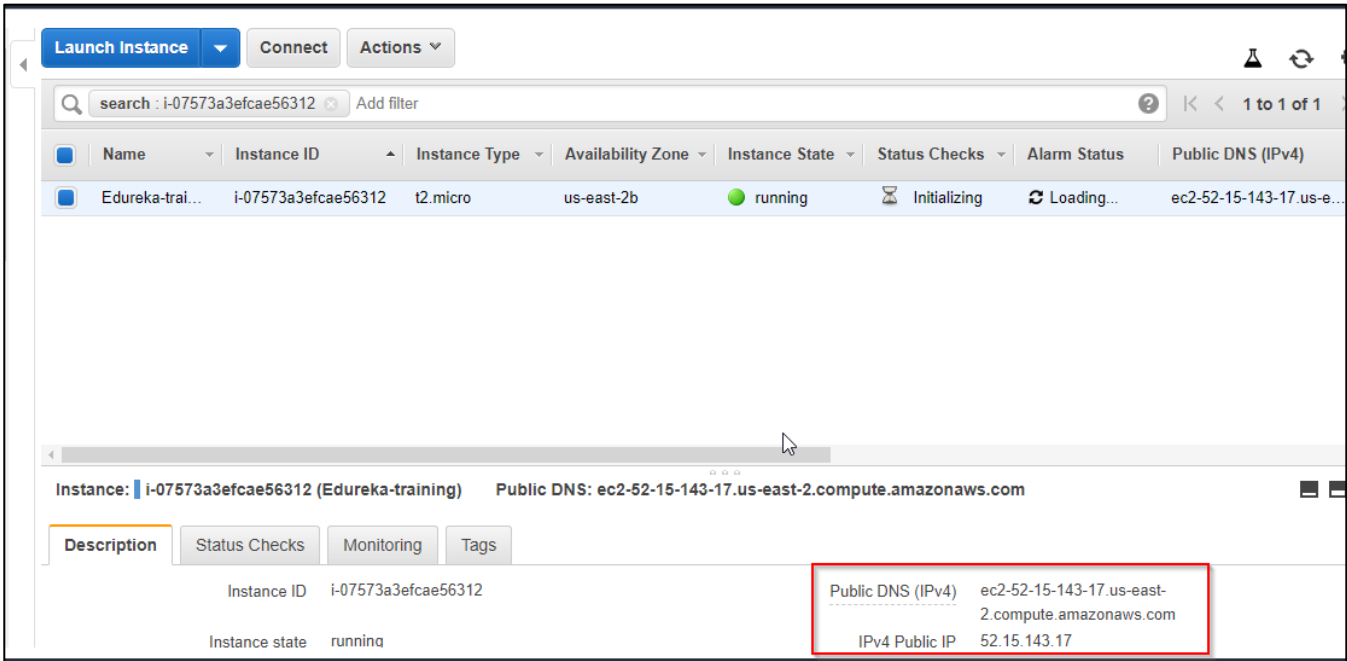
Key pair name

Download Key Pair

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

[Cancel](#) **Launch Instances**

Step 11: Get **Public IP or DNS name** to connect it through internet. Also note the Instance Id to connect through CLI.



Step to connect to your Instance:

Step 1: Connect to your EC2 instance from SSH terminal.

- Launch Ubuntu instance in your console
- For Linux, Unix and Mac users -- **.pem** key pair can be used directly
- For Window users -- Convert **.pem** file to PPK file using PuTTYgen software

The different usernames for the AMIs are:

AMI Type	User Name
Linux AMI	ec2-user
Centos	centos
Debian	admin or root
<u>Feroda</u>	ec2-user or <u>feroda</u>
RHEL	ec2-user or root
SUSE	ec2-user or root
Ubuntu	ubuntu or root
Custom AMI	Check with AMI provider

- The default username for a Ubuntu machine is **ubuntu**

Use the command prompt to connect to your EC2 instance and type the following code.

```
ssh -i <.pem file Absolute path> username@public-IP
```

```
ssh -i C:/Users/Edureka/Downloads/Key.pem ubuntu@13.1.53.42
```

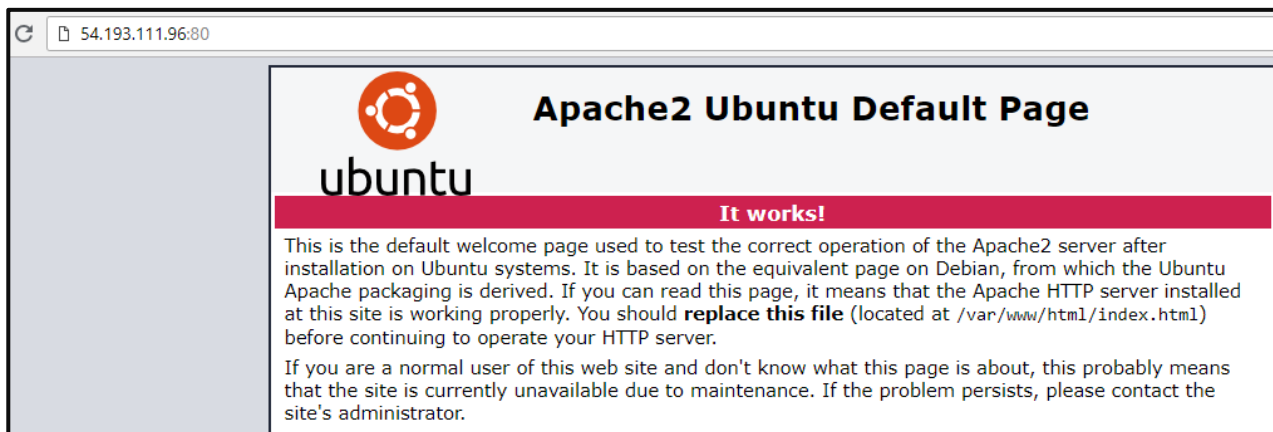
Step 2: Install your Apache server.

- Install your Apache server by typing the below code

```
sudo apt-get update
```

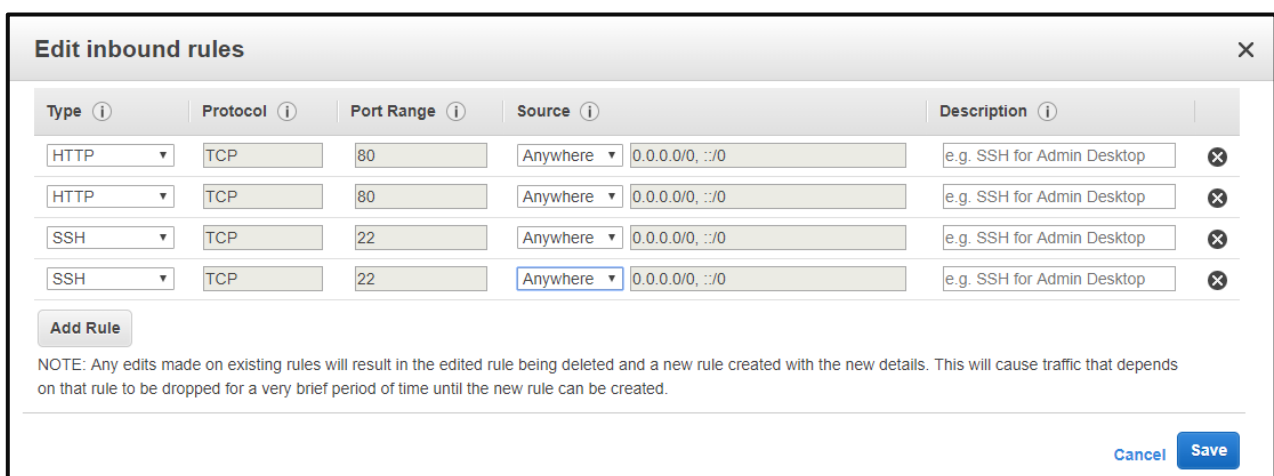
```
sudo apt-get install apache2
```

Step3: Test whether your apache server is running, by typing `http://Public IP address:80` in your browser.



Step4: Troubleshoot.

- Check your security group, if you are not able to connect to the EC2 from the Internet



Step5: Create a Website.

- Create the directory

```
sudo mkdir -p /var/www/edureka123456.tk/public_html
```

- To give permissions to the folder you created, type the following code

```
sudo chmod -R 755 /var/www
```

- To change the permissions, type the following code

```
sudo chown -R $USER:$USER /var/www/edureka123456.tk/public_html
```

```
ubuntu@ip-172-31-4-153:~$ sudo chown -R $USER:$USER /var/www/edureka123456.com/public_html
ubuntu@ip-172-31-4-153:~$ sudo chmod -R 755 /var/www
```

- To enter the directory you have created, type the following code

```
cd /var/www/edureka123456.tk/public_html
```

- To create an index.html file, type the following code

```
nano /var/www/edureka123456.tk/public_html/index.html
```

```
<!DOCTYPE html>
<html>
<head>
<style>
.a{  background-color: #2471A3;
color: white;
padding: 12px 20px;
border: none;
border-radius: 4px;
cursor: pointer;
float: center;  }
```

```
.bg {background-image: url("https://bit.ly/2OEVTYp");
/* Full height */ height: 100%;
background-position: center;
background-repeat: no-repeat;
background-size: cover; }
.label {
    color: white;
    padding: 8px;
    font-family: Arial;
}
</style>
</head>
<body class="bg" style="padding: 210px 0; background-color:
#dbfcf9;">
<center>
    <h3><font size="24"> <font color="white">Welcome to edureka
website</font></h3>
</center>
</body>
</html>
```

Step6: Create a Host file.

- To create a host file, type the below code

```
cd /etc/apache2/sites-available/
```

```
sudo cp /etc/apache2/sites-available/000-default.conf  
/etc/apache2/sites-available/edureka123456.tk.conf
```

```
ubuntu@ip-172-31-6-148:/var/www/edureka123456.tk/public_html$ cd /etc/apache2/si  
tes-available/  
ubuntu@ip-172-31-6-148:/etc/apache2/sites-available$ sudo cp /etc/apache2/sites-  
available/000-default.conf /etc/apache2/sites-available/edureka123456.tk.conf
```

- Edit the virtual host file as

```
sudo nano /etc/apache2/sites-available/edureka123456.tk.conf
```

- Type the code in it

```
<VirtualHost *:80>  
  
ServerAdmin info@edureka123456.tk  
ServerName edureka123456.tk  
ServerAlias www.edureka123456.tk  
DocumentRoot /var/www/edureka123456.tk/public_html  
ErrorLog ${APACHE_LOG_DIR}/error.log  
CustomLog ${APACHE_LOG_DIR}/access.log combined  
  
</VirtualHost>
```

Step7: Enable the host file.

- To enable the host file, type the following code

```
sudo a2ensite edureka123456.tk.conf
```

```
GNU nano 2.9.3
VirtualHost *:80>
ServerAdmin info@edurekal23456.com
ServerName edurekal23456.com
ServerAlias www.edurekal23456.com
DocumentRoot /var/www/edurekal23456.com/public_html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

- To disable the default host file, type the following code

```
sudo a2dissite 000-default.conf
```

Step8: Restart the Apache server.

```
sudo service apache2 restart
```

Step9: Point your local host file to the website you have created.

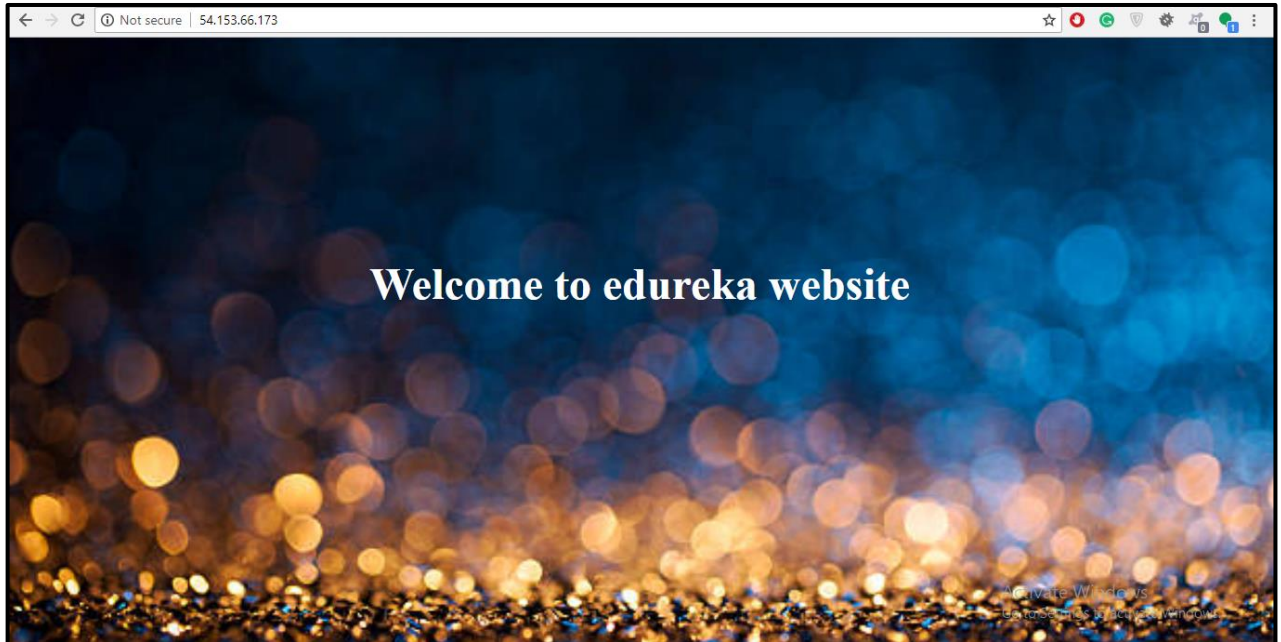
- Type the below code

```
sudo nano /etc/hosts
```

```
Public_IP_address edureka123456.tk
```

```
GNU nano 2.9.3
127.0.0.1 localhost
54.153.66.173 edureka123456.tk
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

- Now when you enter your Public IP address in your web browser, you will be able to see your hosted website



Conclusion

You have successfully configured an EC2 instance and hosted a website via AWS EC2 instance.