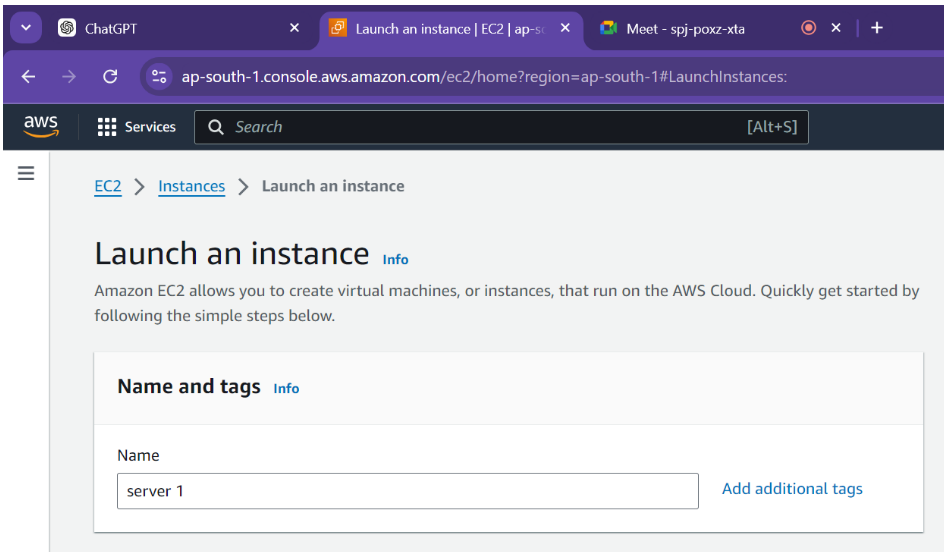
Demo : Hosting multiple websites on a single web server using EC2 Instance

1. Launch instance :

- OS- ubuntu server 22.04 LTS
- Instance type: t2.micro
- Key pair: mykey1
- Create security group: SSH, HTTP

-> Launch Instance



ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

EC2 > Instances > Launch an instance

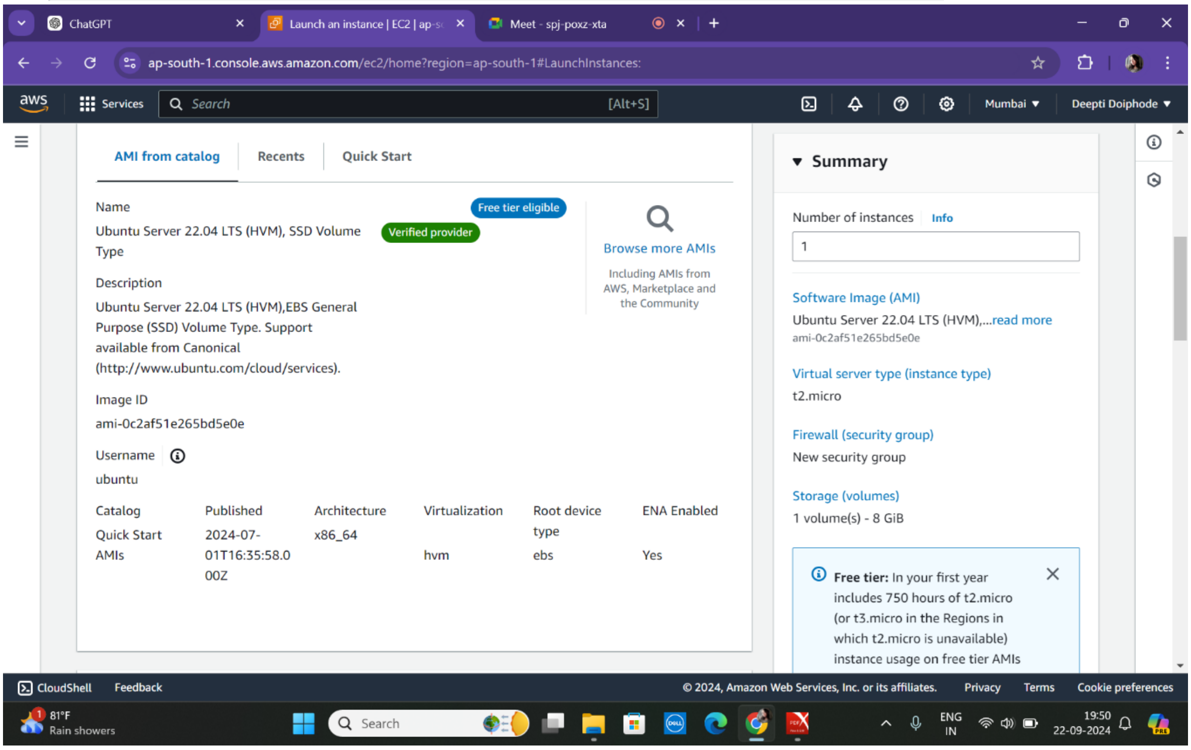
Launch an instance Info

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: server 1

Add additional tags



AMI from catalog | Recents | Quick Start

Name: Ubuntu Server 22.04 LTS (HVM), SSD Volume Type — Free tier eligible — Verified provider

Description: Ubuntu Server 22.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Image ID: ami-0c2af51e265bd5e0e

Username: ubuntu

Catalog	Published	Architecture	Virtualization	Root device type	ENA Enabled
Quick Start AMIs	2024-07-01T16:35:58.000Z	x86_64	hvm	ebs	Yes

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

Summary

Number of instances Info: 1

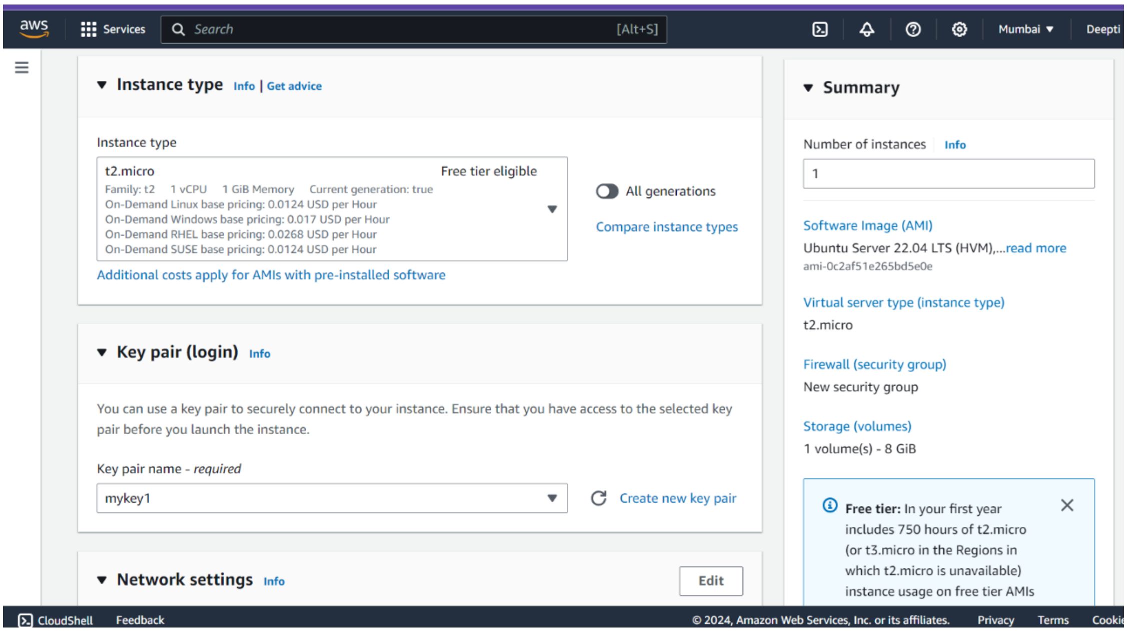
Software Image (AMI): Ubuntu Server 22.04 LTS (HVM),...read more
ami-0c2af51e265bd5e0e

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs



▼ **Instance type** Info | Get advice

Instance type

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0268 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

▼ **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*

mykey1

Create new key pair

▼ **Network settings** Info Edit

▼ **Summary**

Number of instances Info

1

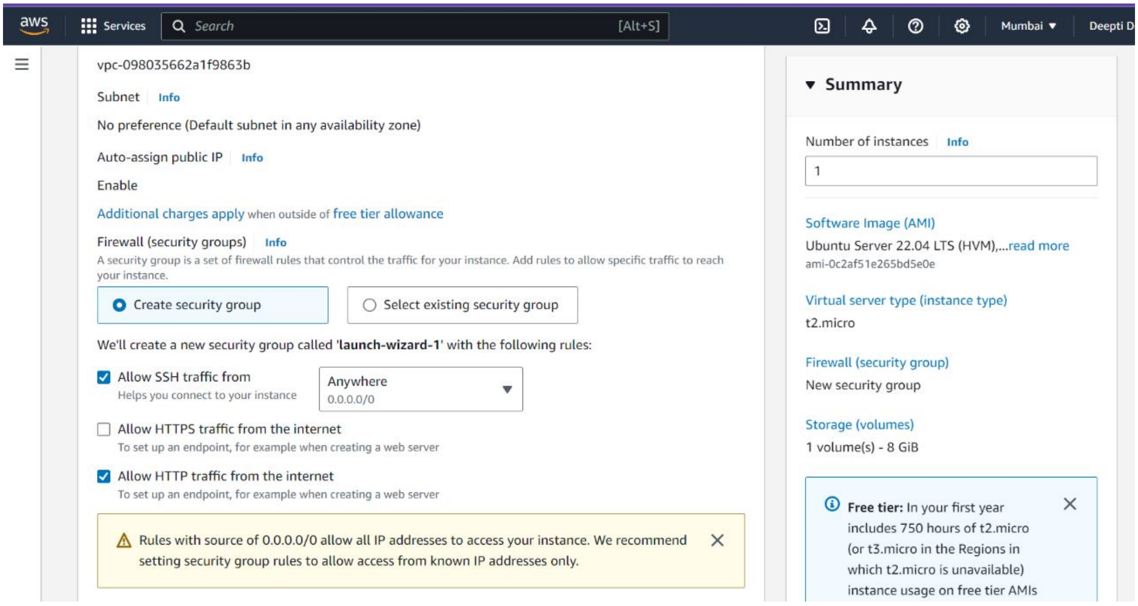
Software Image (AMI)
Ubuntu Server 22.04 LTS (HVM),...read more
ami-0c2af51e265bd5e0e

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs



vpc-098035662a1f9863b

Subnet Info

No preference (Default subnet in any availability zone)

Auto-assign public IP Info

Enable

Additional charges apply when outside of free tier allowance

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.

● Create security group ○ Select existing security group

We'll create a new security group called '**launch-wizard-1**' with the following rules:

☑ Allow SSH traffic from
Helps you connect to your instance
Anywhere
0.0.0.0/0

□ Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

☑ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ 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.

▼ **Summary**

Number of instances Info

1

Software Image (AMI)
Ubuntu Server 22.04 LTS (HVM),...read more
ami-0c2af51e265bd5e0e

Virtual server type (instance type)
t2.micro

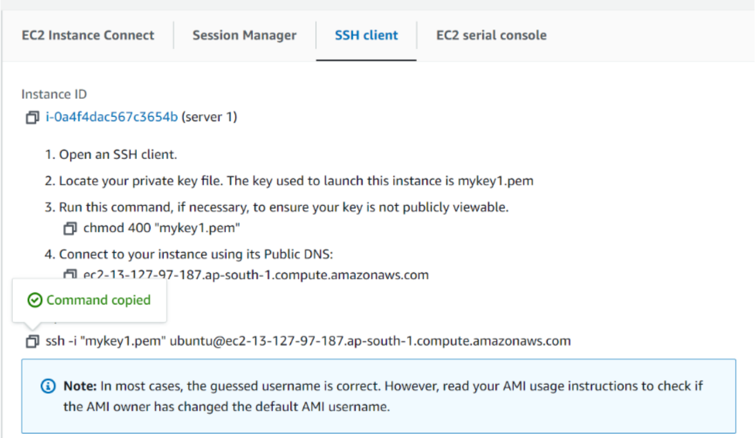
Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs

➔ Instance id ->connect ->SSH client

(copy the example below: ssh -i "mykey1.pem" ubuntu@ec2-13-127-97-187.ap-south-1.compute.amazonaws.com)



Instance ID

i-0a4f4dac567c3654b (server 1)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is mykey1.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.

```
chmod 400 "mykey1.pem"
```

4. Connect to your instance using its Public DNS:

```
ec2-13-127-97-187.ap-south-1.compute.amazonaws.com
```

✓ Command copied

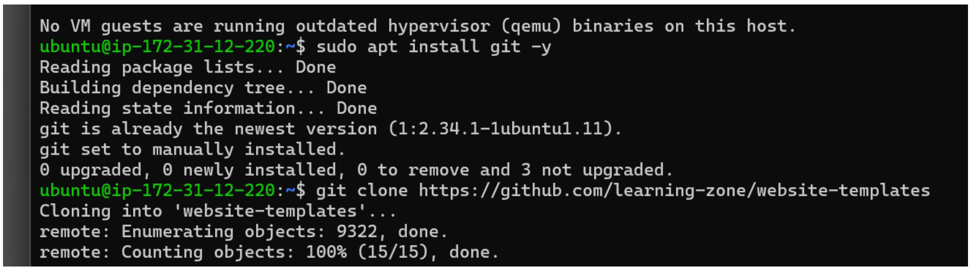
```
ssh -i "mykey1.pem" ubuntu@ec2-13-127-97-187.ap-south-1.compute.amazonaws.com
```

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

2. Go to Command Prompt :

➔ And enter commands =

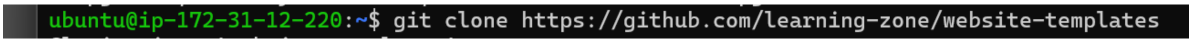
- cd downloads
  (paste ssh protocol example here)
- Yes
- sudo apt update -y (command – to update instance)
- sudo apt upgrade -y (command – to upgrade instance)
  (Enter)
- sudo apt install apache2 -y (install apache web server)
  (Enter)
- sudo apt install git -y



```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-12-220:~$ sudo apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.34.1-1ubuntu1.11).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
ubuntu@ip-172-31-12-220:~$ git clone https://github.com/learning-zone/website-templates
Cloning into 'website-templates'...
remote: Enumerating objects: 9322, done.
remote: Counting objects: 100% (15/15), done.
```

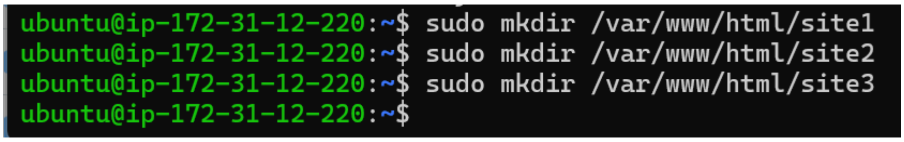
(Enter)

- git clone https://github.com/learning-zone/website-templates (to clone repository)



```
ubuntu@ip-172-31-12-220:~$ git clone https://github.com/learning-zone/website-templates
```

- sudo mkdir /var/www/html/site1 (to create directory for each site)
- sudo mkdir /var/www/html/site2
- sudo mkdir /var/www/html/site3



```
ubuntu@ip-172-31-12-220:~$ sudo mkdir /var/www/html/site1
ubuntu@ip-172-31-12-220:~$ sudo mkdir /var/www/html/site2
ubuntu@ip-172-31-12-220:~$ sudo mkdir /var/www/html/site3
ubuntu@ip-172-31-12-220:~$
```

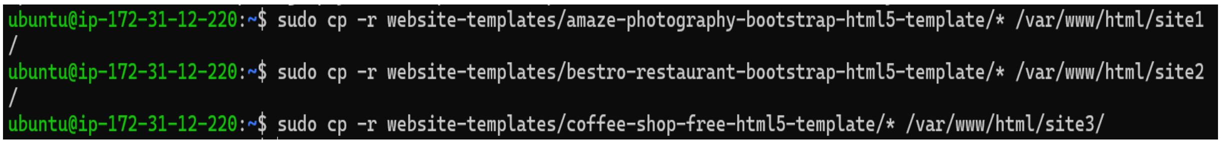
➔ Go to github and select a template -> Copy path of template

➔ paste in this command in command prompt:

➔ we have taken 3 website templates below: photography website
Restaurant website
Coffee shop website

(in command : sudo cp -r website-tempates/ link of website/* /var/www/html/site/ this is needed)

sudo cp -r website-templates/amaze-photography-bootstrap-html5-template/* /var/www/html/site1/
sudo cp -r website-templates/ bestro-restaurant-bootstrap-html5-template /var/www/html/site2/
sudo cp -r website-templates/coffee-shop-free-html5-template/* /var/www/html/site3/

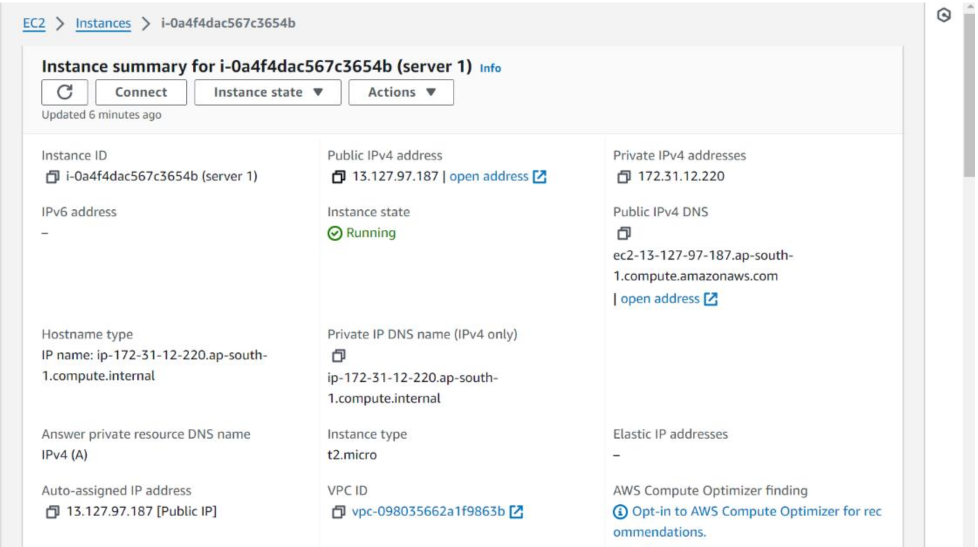


```
ubuntu@ip-172-31-12-220:~$ sudo cp -r website-templates/amaze-photography-bootstrap-html5-template/* /var/www/html/site1
/
ubuntu@ip-172-31-12-220:~$ sudo cp -r website-templates/bestro-restaurant-bootstrap-html5-template/* /var/www/html/site2
/
ubuntu@ip-172-31-12-220:~$ sudo cp -r website-templates/coffee-shop-free-html5-template/* /var/www/html/site3/
```

- sudo systemctl restart apache2

3. Instance

->copy public IPv4 id (13.127.97.187) & paste on Browser



Instance summary for i-0a4f4dac567c3654b (server 1) Info

Updated 6 minutes ago

Instance ID
i-0a4f4dac567c3654b (server 1)

Public IPv4 address
13.127.97.187 | open address

Private IPv4 addresses
172.31.12.220

IPv6 address
–

Instance state
Running

Public IPv4 DNS
ec2-13-127-97-187.ap-south-1.compute.amazonaws.com
| open address

Hostname type
IP name: ip-172-31-12-220.ap-south-1.compute.internal

Private IP DNS name (IPv4 only)
ip-172-31-12-220.ap-south-1.compute.internal

Answer private resource DNS name
IPv4 (A)

Instance type
t2.micro

Elastic IP addresses
–

Auto-assigned IP address
13.127.97.187 [Public IP]

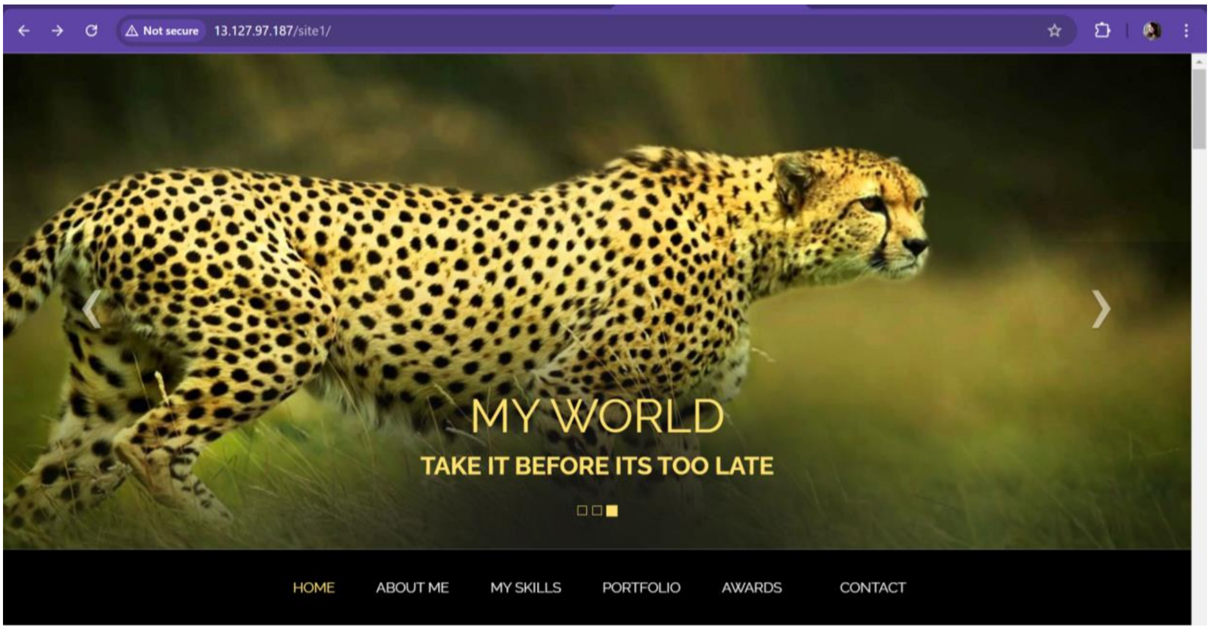
VPC ID
vpc-098035662a1f9863b

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations.

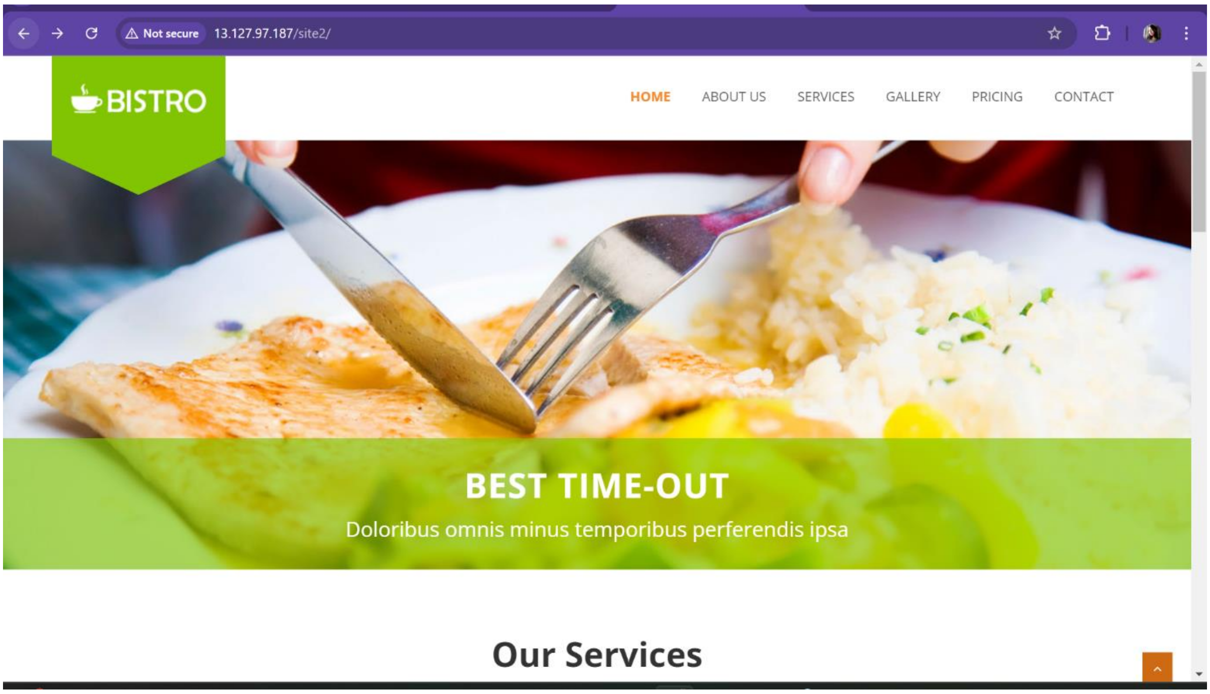
-> now edit the ip address entered in the browser:

13.127.97.187/site1/
13.127.97.187/site2/
13.127.97.187/site3/

SITE 1



SITE 2

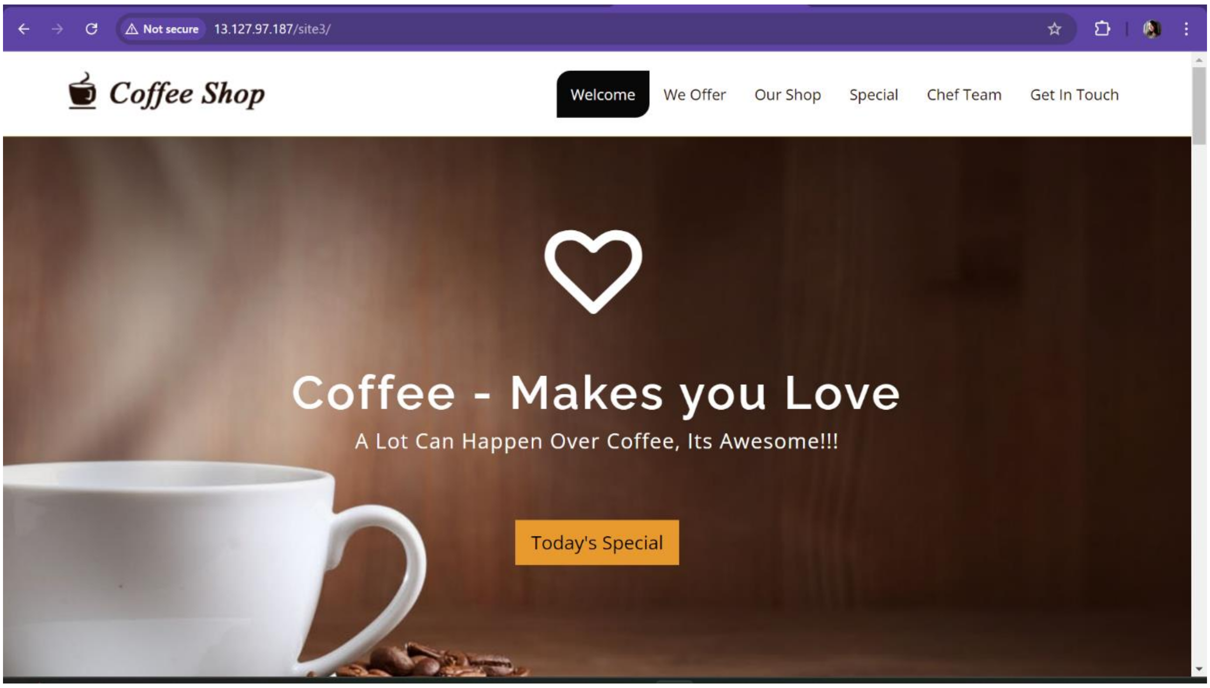


BEST TIME-OUT

Doloribus omnis minus temporibus perferendis ipsa

Our Services

SITE 3



Coffee - Makes you Love

A Lot Can Happen Over Coffee, Its Awesome!!!