

# ***1. Describe IaaS.***

**Infrastructure as a Service (IaaS)** is a form of cloud computing that provides virtualized computing resources over the Internet. IaaS is one of the three main categories of cloud services, alongside Software as a Service (SaaS) and Platform as a Service (PaaS). In an IaaS model, a third-party provider hosts hardware, software, servers, storage, and other infrastructure components on behalf of its users.

## **Key Features of IaaS:**

**Virtual Machines (VMs):** Users can rent VMs for running their applications.

**Scalability:** Resources can be scaled up or down based on demand.

**Pay-as-you-go Pricing:** Users pay only for the resources they consume.

**Automated Administrative Tasks:** Including backup, security, and disaster recovery.

**Multi-tenancy:** Multiple users share the same infrastructure.

# ***2. Compute and Storage Services in AWS and GCP.***

Amazon Web Services (AWS)

## **Compute Services:**

**Amazon EC2 (Elastic Compute Cloud):** Virtual services in the cloud

**Amazon ECS (Elastic Container Service):** Run and manage Docker containers.

**AWS Lambda:** Serverless computing service to run code without provisioning servers

**Amazon EKS (Elastic Kubernetes Service):** Managed Kubernetes service.

**AWS Fargate:** Serverless compute engine for containers

## **Global Cloud Platform (GCP)**

## **Compute Services:**

**Google Compute Engine (GCE):** Virtual machines running in Google's data centers.

**Google Kubernetes Engine (GKE):** Managed Kubernetes service.

**Google Cloud Functions:** Event-driven serverless compute service.

**Google Cloud Run:** Managed compute platform for running containerized applications

## Storage Services:

**Google Cloud Storage:** Object storage service.

**Persistent Disks:** Block storage for use with GCE-

**Filestore:** Fully-managed file storage.

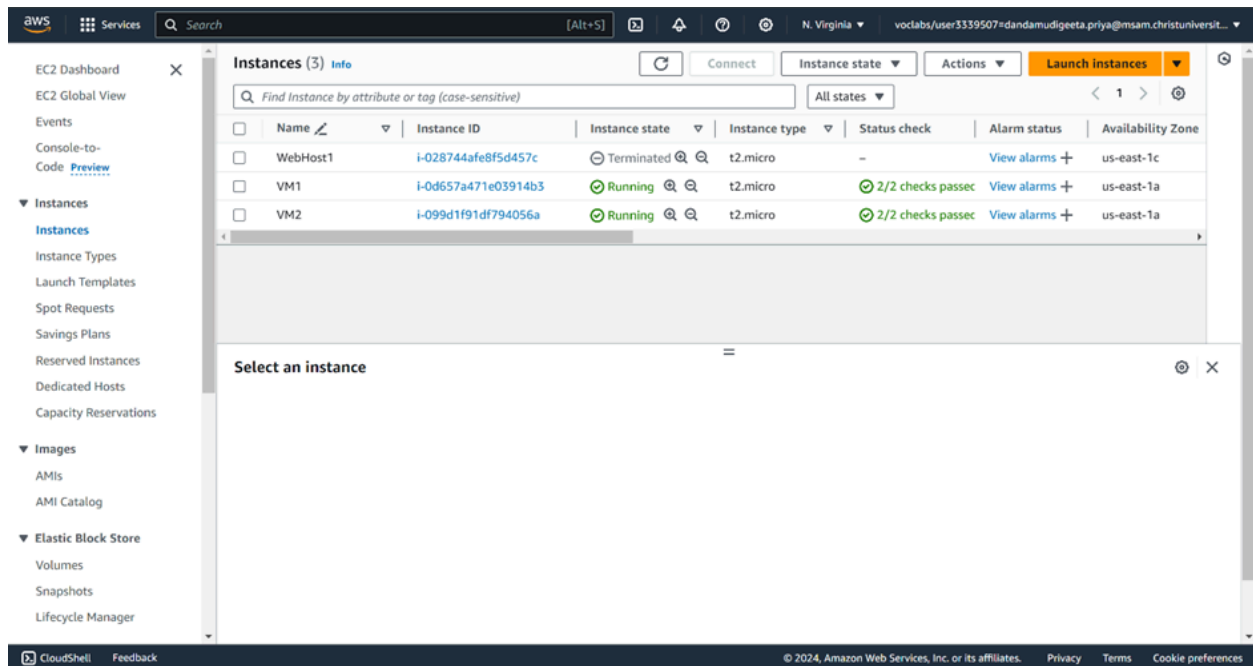
**Google Cloud Storage Nearline and Coldline:** Storage classes for data archiving and long-term backup-

**Transfer Appliance:** Hardware appliance for transferring large amounts of data to Google Cloud.

These services provide a robust and flexible infrastructure for a wide range of applications, from simple websites to complex data analytics.

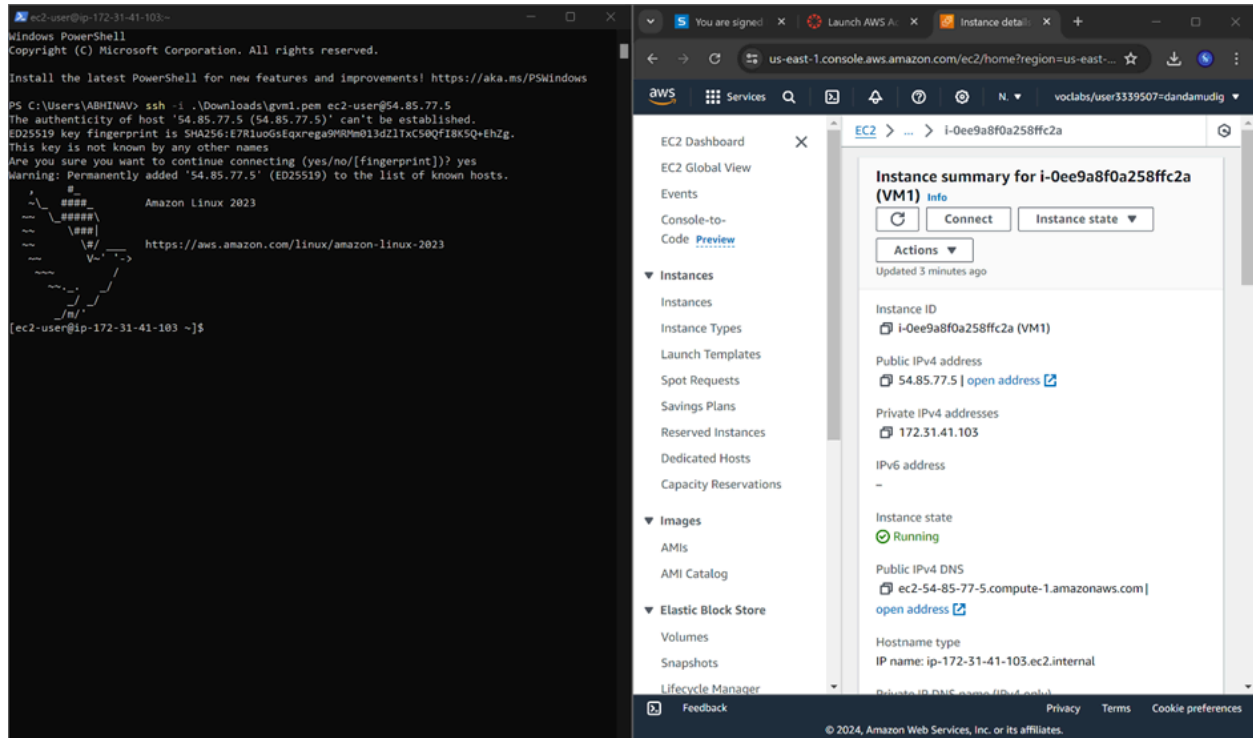
## 3. Creating 2 AWS EC2 Instances.

### 1: Creating the EC2 Instances.



2. Copy the Public IPv4 address and use the Windows SSH Client to connect to the instance VM1 using this command *ssh -i* .

*Downloads\gvm1.pem ec2-user@"ip-address"*



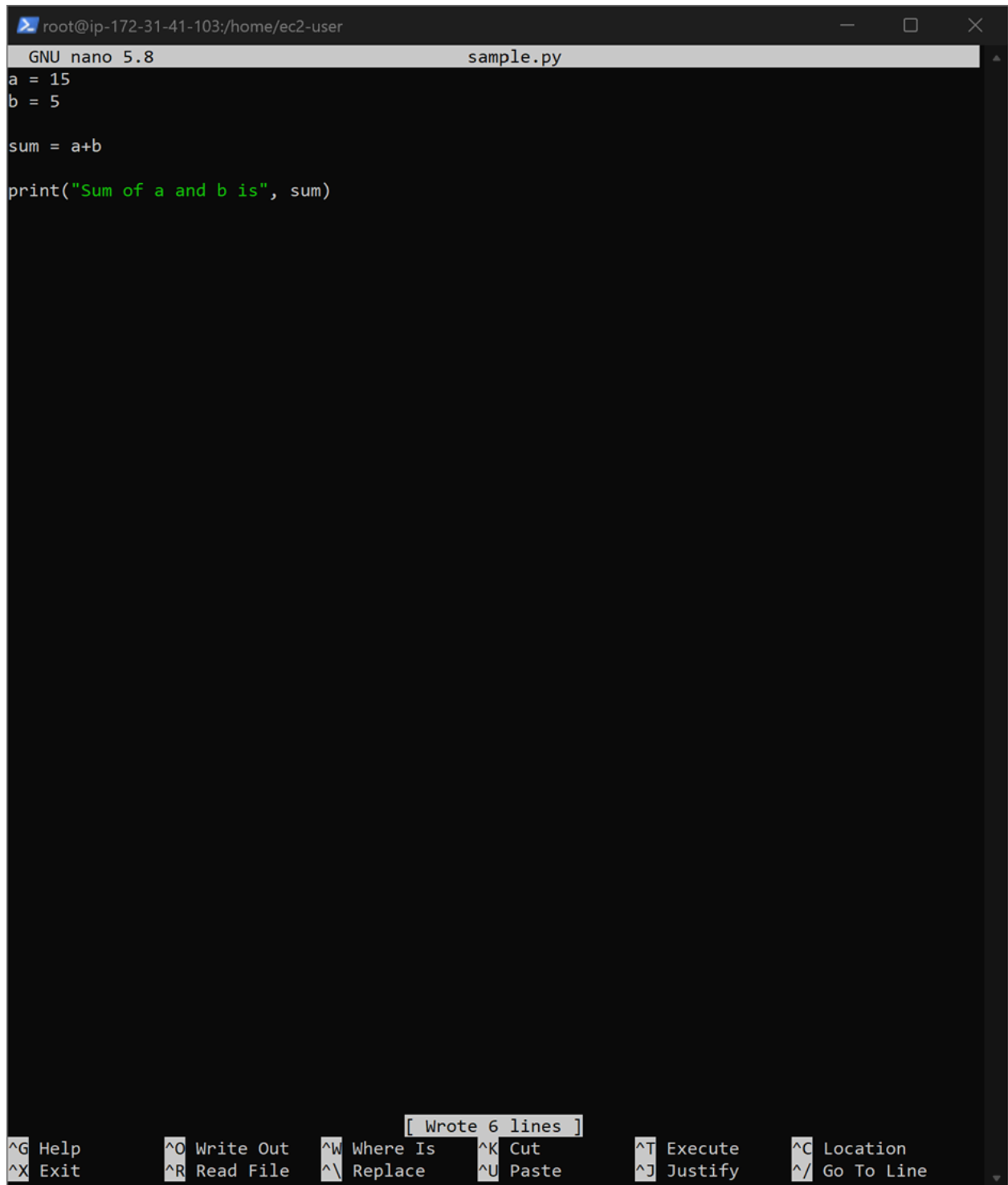
It's standard procedure after this use basic linux commands to install python and run a simple python program

## ***Installing python***

<https://aws.amazon.com/linux/amazon-linux-2023>

```
Complete!  
[root@ip-172-31-41-103 ec2-user]#
```

## *Making a python file and saving it*



```
root@ip-172-31-41-103:/home/ec2-user
GNU nano 5.8 sample.py
a = 15
b = 5

sum = a+b

print("Sum of a and b is", sum)
```

[ Wrote 6 lines ]

<b>^G</b> Help	<b>^O</b> Write Out	<b>^W</b> Where Is	<b>^K</b> Cut	<b>^T</b> Execute	<b>^C</b> Location
<b>^X</b> Exit	<b>^R</b> Read File	<b>^_</b> Replace	<b>^U</b> Paste	<b>^J</b> Justify	<b>^/</b> Go To Line

### *Listing the files and running the python file*

```
root@ip-172-31-41-103:/home/ec2-user  
~  
[ec2-user@ip-172-31-41-103 ~]$ sudo su  
[root@ip-172-31-41-103 ec2-user]# yum update -y  
Last metadata expiration check: 0:07:39 ago on Tue Jul 2 16:53:58 2024.  
Dependencies resolved.  
Nothing to do.  
Complete!  
[root@ip-172-31-41-103 ec2-user]# yum install python  
Last metadata expiration check: 0:07:47 ago on Tue Jul 2 16:53:58 2024.  
Dependencies resolved.  
  
=====
```

Package	Arch	Version	Repository	Size
---------	------	---------	------------	------

```
=====
```

Installing:

python-unversioned-command	noarch	3.9.16-1.amzn2023.0.8	amazonlinux	10 k
----------------------------	--------	-----------------------	-------------	------

Transaction Summary

```
=====
```

Install 1 Package

Total download size: 10 k  
Installed size: 23  
Is this ok [y/N]: y  
Downloading Packages:  
python-unversioned-command-3.9.16-1.amzn2023.0.8.noarch.rpm 164 kB/s | 10 kB 00:00  
-----  
Total 83 kB/s | 10 kB 00:00

Running transaction check  
Transaction check succeeded.  
Running transaction test  
Transaction test succeeded.  
Running transaction  
Preparing : 1/1  
Installing : python-unversioned-command-3.9.16-1.amzn2023.0.8.noarch 1/1  
Running scriptlet: python-unversioned-command-3.9.16-1.amzn2023.0.8.noarch 1/1  
Verifying : python-unversioned-command-3.9.16-1.amzn2023.0.8.noarch 1/1

Installed:  
python-unversioned-command-3.9.16-1.amzn2023.0.8.noarch

Complete!

```
[root@ip-172-31-41-103 ec2-user]# nano sample.py  
[root@ip-172-31-41-103 ec2-user]# la  
bash: la: command not found  
[root@ip-172-31-41-103 ec2-user]# ls  
sample.py  
[root@ip-172-31-41-103 ec2-user]# python sample.py  
Sum of a and b is 20  
[root@ip-172-31-41-103 ec2-user]#
```

*Installing Apache and enabling it.*

*Start the server using: systemctl start httpd*

*Enable the server using: systemctl enable httpd*

```
root@ip-172-31-41-103:/home/ec2-user
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                1/12
  Installing     : apr-1.7.2-2.amzn2023.0.2.x86_64 1/12
  Installing     : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64 2/12
  Installing     : apr-util-1.6.3-1.amzn2023.0.1.x86_64 3/12
  Installing     : mailcap-2.1.49-3.amzn2023.0.3.noarch 4/12
  Installing     : httpd-tools-2.4.59-2.amzn2023.x86_64 5/12
  Installing     : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 6/12
  Running scriptlet: httpd-filesystem-2.4.59-2.amzn2023.noarch 7/12
  Installing     : httpd-filesystem-2.4.59-2.amzn2023.noarch 7/12
  Installing     : httpd-core-2.4.59-2.amzn2023.x86_64 8/12
  Installing     : mod_http2-2.0.27-1.amzn2023.0.2.x86_64 9/12
  Installing     : mod_lua-2.4.59-2.amzn2023.x86_64 10/12
  Installing     : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 11/12
  Installing     : httpd-2.4.59-2.amzn2023.x86_64 12/12
  Running scriptlet: httpd-2.4.59-2.amzn2023.x86_64 12/12
  Verifying      : apr-1.7.2-2.amzn2023.0.2.x86_64 1/12
  Verifying      : apr-util-1.6.3-1.amzn2023.0.1.x86_64 2/12
  Verifying      : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64 3/12
  Verifying      : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 4/12
  Verifying      : httpd-2.4.59-2.amzn2023.x86_64 5/12
  Verifying      : httpd-core-2.4.59-2.amzn2023.x86_64 6/12
  Verifying      : httpd-filesystem-2.4.59-2.amzn2023.noarch 7/12
  Verifying      : httpd-tools-2.4.59-2.amzn2023.x86_64 8/12
  Verifying      : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 9/12
  Verifying      : mailcap-2.1.49-3.amzn2023.0.3.noarch 10/12
  Verifying      : mod_http2-2.0.27-1.amzn2023.0.2.x86_64 11/12
  Verifying      : mod_lua-2.4.59-2.amzn2023.x86_64 12/12

Installed:
apr-1.7.2-2.amzn2023.0.2.x86_64
apr-util-1.6.3-1.amzn2023.0.1.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-2.4.59-2.amzn2023.x86_64
httpd-core-2.4.59-2.amzn2023.x86_64
httpd-filesystem-2.4.59-2.amzn2023.noarch
httpd-tools-2.4.59-2.amzn2023.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_http2-2.0.27-1.amzn2023.0.2.x86_64
mod_lua-2.4.59-2.amzn2023.x86_64

Complete!
[root@ip-172-31-41-103 ec2-user]# systemctl start httpd
[root@ip-172-31-41-103 ec2-user]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-41-103 ec2-user]#
```



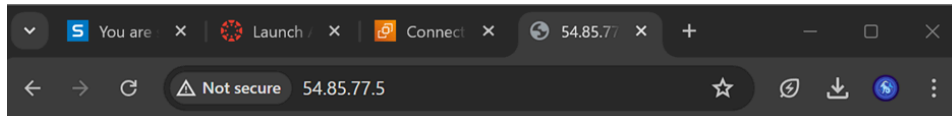
### *Writing the html code.*

```
ec2-user@ip-172-31-24-131:~  
Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
  
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows  
  
PS C:\Users\ABHINAV> ssh  
usage: ssh [-46AaCfGgKkMnNqsTtVvXxYy] [-B bind_interface]  
          [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]  
          [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]  
          [-i identity_file] [-J [user@]host[:port]] [-L address]  
          [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]  
          [-Q query_option] [-R address] [-S ctl_path] [-W host:port]  
          [-w local_tun[:remote_tun]] destination [command]  
  
PS C:\Users\ABHINAV> ssh -i .\Downloads\vm2.pem ec2-user@54.160.234.178  
  
#_  
~\_#####_ Amazon Linux 2023  
~~ \#####\  
~~ \###|  
~~ \#/_____  
~~ V~' '->  
~~~~  
~~~~_._.  
~~/_/_/_/  
~/m/'
```

```
Last login: Sun Jun 30 06:19:58 2024 from 103.105.225.66  
[ec2-user@ip-172-31-24-131 ~]$ sudo yum update -y  
Last metadata expiration check: 1:21:39 ago on Sun Jun 30 05:08:34 2024.  
Dependencies resolved.  
Nothing to do.  
Complete!  
[ec2-user@ip-172-31-24-131 ~]$ sudo yum install httpd  
Last metadata expiration check: 1:21:47 ago on Sun Jun 30 05:08:34 2024.  
Package httpd-2.4.59-2.amzn2023.x86_64 is already installed.  
Dependencies resolved.  
Nothing to do.  
Complete!  
[ec2-user@ip-172-31-24-131 ~]$ echo "<html><body><h1>Hello World!\</h1></body></html>" | sudo  
tee /var/www/html/index.html  
<html><body><h1>Hello World!\</h1></body></html>  
[ec2-user@ip-172-31-24-131 ~]$
```

**4. Configure a Webserver on ‘Regno\_EC2\_VM2’ Instance and host your organization's website (Static Website) and provide access only to your machine.**

*Displaying the webpage: go to the instance page and copy the public IPv4 address and copy paste it into a new tab and the web page should load.*



**Hello. This is Christ University**