

2147130 – CC LAB 5

Q) 1. Describe STaaS.

ANS – It stands for storage as a service. STaaS can be used for a variety of purposes, from long-term archival storage to short-term transfers of large amounts of data. Cloud-based STaaS also offers customers a wide range of functionality and additional services that can be performed without the expertise of a storage engineer. These include disaster recovery, data backup, data storage, bulk data transferred, block storage, and SSD storage to name a few.

Q) 2. Mention the different storage types offered by the cloud providers and list the example services for each type in GCP/AWS/Azure.

ANS -

Types of AWS Storage Services –

1. Simple Storage Service (S3)

Amazon S3 is an object storage service that stores data of any type and size. It can store data for any business such as web applications, mobile applications, backup, archive, and analytics. It also provides easy access control management for all your specific requirements and is almost 100% durable it also allows a simple web-based file explorer to upload files, create folders or delete them.

2. Elastic Block Storage (EBS) –

EBS provides block storage which is similar to hard drives to store any kind of data persistently. This can be attached to any EC2 instance and used as block storage, which even allows you to install any operating system. EBS volumes are placed in availability zones so that they are replicated to prevent loss of data due to single component failures. They provide absolute low-latency performance and you can also scale up or down your resources as and when required. EBS is available in both SSD and HDD formats depending on your requirement for speed and volume.

3. Elastic File System (EFS)

EFS is a managed network file system that is easy to set up right from the amazon console or CLI. When you have multiple EC2 instances needed to access the same file system EFS helps in providing just that. Unlike EBS, EFS is built using the NFS4.x protocol on SSDs and have a much faster throughput. This also means that EFS is much more expensive than EBS as it can be used on very large analytical workloads.

4. Amazon FSx for Lustre

Luster is a file system used for compute-intensive workloads. This mainly comes into the picture when you run machine learning operations on large data sets or when you need to run media encoding workloads. Running Lustre separately requires a lot of expertise in setting it up and configuring it for the right workloads. With the help of Amazon FSx, this can be avoided and a simple interface on the console helps you to quickly get started and start working on your data.

5. Amazon S3 Glacier

The glacier is used mainly for archival and long-term data storage. This means that there is a low retrieval rate on this storage system due to which it is offered at an extremely cheap rate. It does also come with compliant security features to encrypt your data. Glacier allows you to run queries and analytics on it directly and you will be charged only for the few minutes or hours when you read the data.

6. Amazon FSx for Windows File Server

Whenever you need to run your windows specific software that needs to access the proprietary windows file system on the cloud, AWS provides you with Amazon FSx to easily achieve that. Windows-based .Net applications, ERPs and CRMs require shared file storage to move workloads between them.

7. AWS Storage Gateway

Storage Gateway is a simple way to let your on-premise applications store, access, or archive the data into the AWS cloud. This is achieved by running on a hypervisor on one of the machines in your data center which contains the storage gateway and then is available on AWS to connect to S3, Glacier or EBS. It provides a highly optimized, network resilient, and low-cost way to move your data from on-prem to the cloud.

Types of GCP Storage Services –

Storage Class	Name for APIs and CLIs	<u>Minimum storage duration</u>	<u>Retrieval fees</u>	Typical monthly availability ¹
Standard storage	STANDARD	None	None	<ul style="list-style-type: none">• >99.99% in multi-regions and dual-regions
Nearline storage	NEARLINE	30 days	Yes	<ul style="list-style-type: none">• 99.99% in regions• 99.95% in multi-regions and dual-regions
Coldline storage	COLDLINE	90 days	Yes	<ul style="list-style-type: none">• 99.9% in regions• 99.95% in multi-regions and dual-regions
Archive storage	ARCHIVE	365 days	Yes	<ul style="list-style-type: none">• 99.9% in regions• 99.95% in multi-regions and dual-regions
				<ul style="list-style-type: none">• 99.9% in regions

- Standard storage

Standard storage is best for data that is frequently accessed ("hot" data) and/or stored for only brief periods of time. When used in a region, Standard storage is appropriate for storing data in the same location as Google Kubernetes Engine clusters or Compute Engine instances that use the data. Co-locating your resources maximizes the performance for data-intensive computations and can reduce network charges.

- Nearline storage

Nearline storage is a low-cost, highly durable storage service for storing infrequently accessed data. Nearline storage is a better choice than Standard storage in scenarios where slightly lower availability, a 30-day minimum storage duration, and costs for data access are acceptable trade-offs for lowered at-rest storage costs. Nearline storage is ideal for data you plan to read or modify on average once per month or less. For

example, if you want to continuously add files to Cloud Storage and plan to access those files once a month for analysis, Nearline storage is a great choice.

- Coldline storage

Coldline storage is a very-low-cost, highly durable storage service for storing infrequently accessed data. Coldline storage is a better choice than Standard storage or Nearline storage in scenarios where slightly lower availability, a 90-day minimum storage duration, and higher costs for data access are acceptable trade-offs for lowered at-rest storage costs.

- Archive storage

Archive storage is the lowest-cost, highly durable storage service for data archiving, online backup, and disaster recovery. Unlike the "coldest" storage services offered by other Cloud providers, your data is available within milliseconds, not hours or days. Like Nearline storage and Coldline storage, Archive storage has a slightly lower availability than Standard storage. Archive storage also has higher costs for data access and operations, as well as a 365-day minimum storage duration. Archive storage is the best choice for data that you plan to access less than once a year.

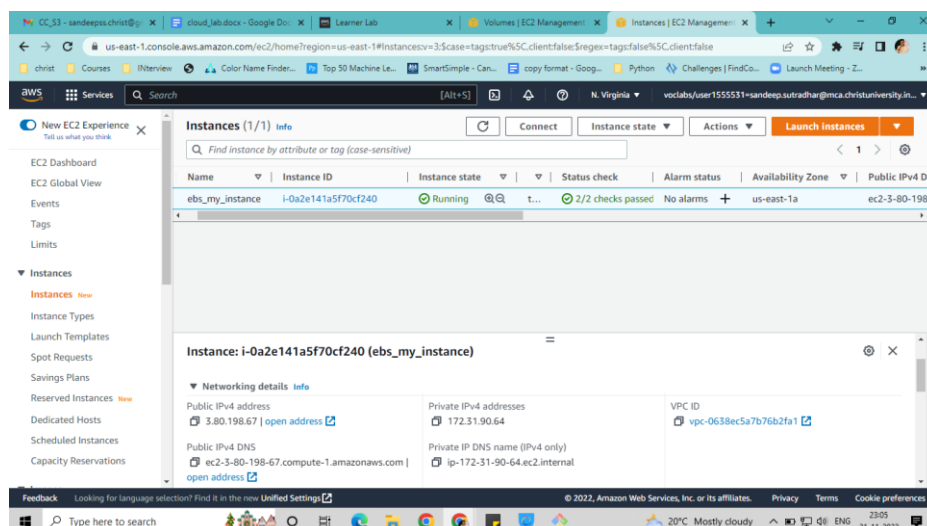
In Microsoft Azure, you can choose two Storage Account types. They are General-Purpose Account and Blob Storage account. General-Purpose Account is designed to operate with all types of Microsoft Azure Storage, except Disk Storage. To create disks inside your Azure Storage, you should first create the Microsoft Azure Virtual Machine. Blob Storage account is designed to work with Block Blobs and Append Blobs. Page Blobs, however, can only be created when you are using a General-Purpose account.

Q) 3. Demonstrate the following

1. Create a New Block Store and attach it to a VM instance.

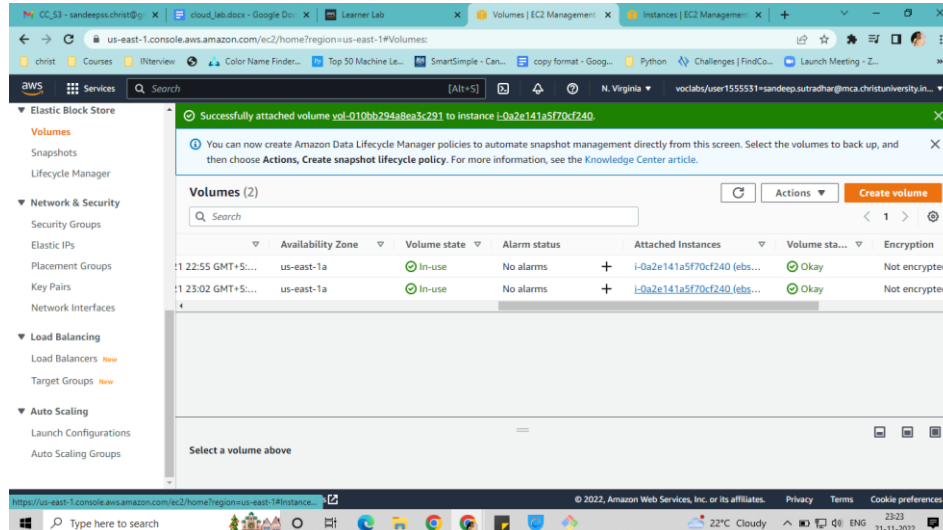
ANS – Following steps can be followed to create a new block storage with mounting (Mounting is a process by which a computer's operating system makes files and directories on a storage device (such as hard drive, CD-ROM, or network share) available for users to access via the computer's file system.)

Step 1 → Create an instance with the key-value pair; → Download the key value pair.



Step 2 → Under the Elastic Block Store go to Volumes → Select create Volume (Already one is there for already created instance) → Select the Volume type (General purpose SSD gp2) → Select Size (GiB) ex – 5GB → Select the availability zone as the instance created zone → Select CREATE VOLUME

Step 3 → Select the volume created (You may give name if you want) → Go to Actions and select attach volumes → Select the instance with which the volume want to add → select attach volume.



Step 4 → Open gitbash go to the folder where .pem file has been downloaded →

Write the following codes

\$ ssh -i 2147130_key_test.pem ec2-user@3.80.198.67 #it is public ip of instance

lsblk

sudo mkfs.ext4 /dev/xvdf

mkdir ssnewdir

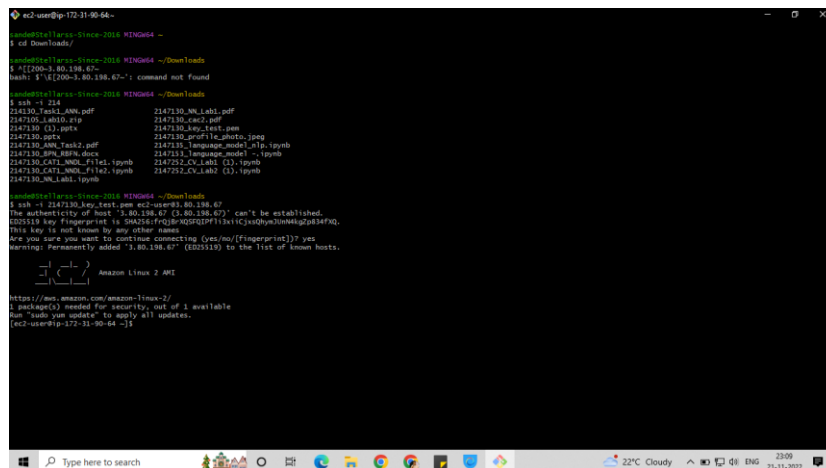
ls -la

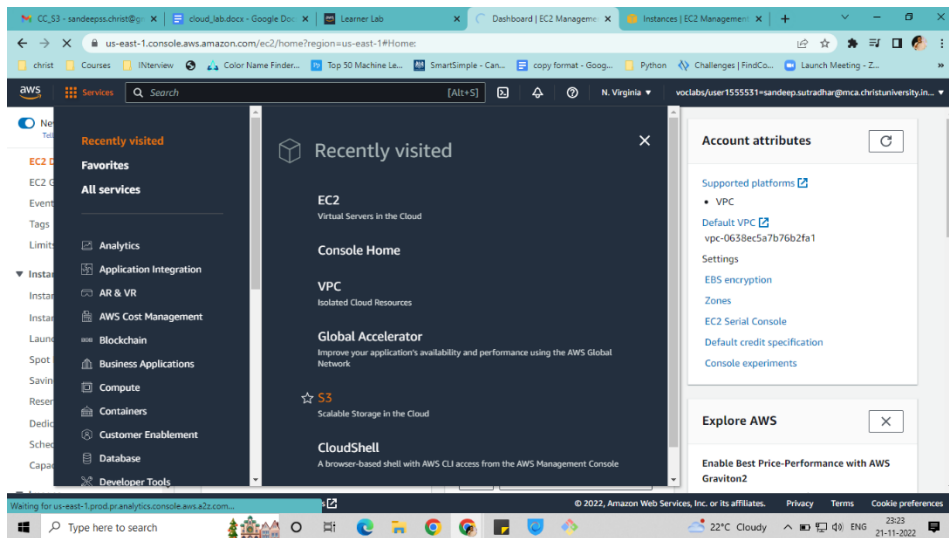
sudo touch test1.txt

sudo vim test1.txt

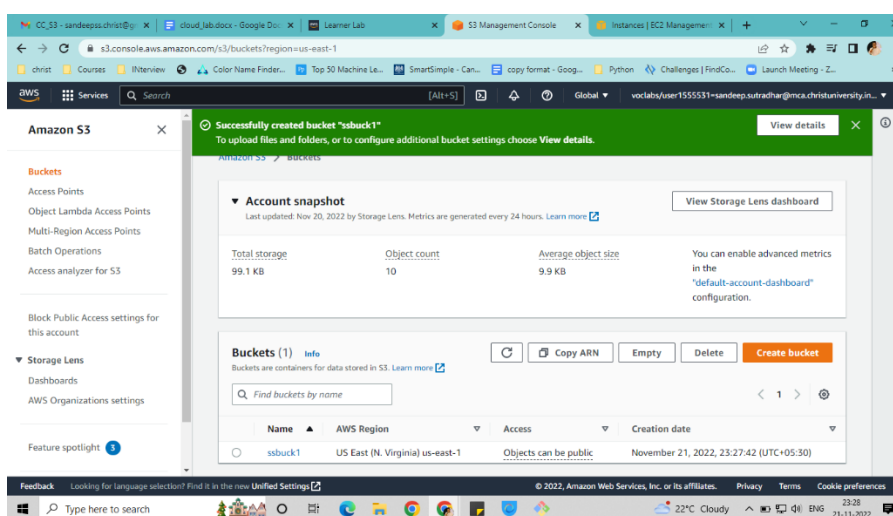
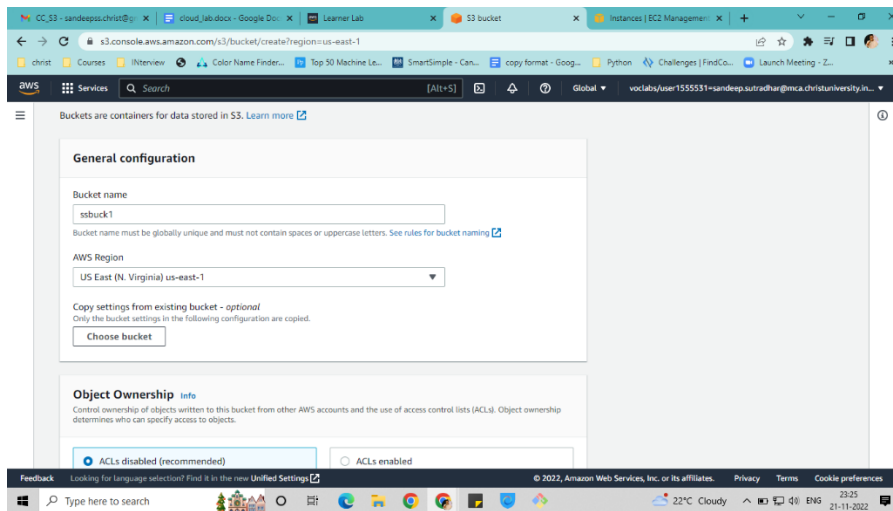
sudo mount /dev/xvdf ssnewdir/
lsblk

Note: if you want to unmount → sudo unmount /ssnewdir/





Click on Buckets → Select Create Bucket → Give Bucket name (ssbuck1) → Select the Aws region → Uncheck the Block all public access → Leave everything as it is → Select Create Bucket.



Step 2 → Install aws cli → open cmd → Write the following codes

Aws configure list → aws configure → give the access key and secret key from the details of aws details → give the region as us-east-1 → enter. → Go to folder C:\Users\sande\aws\credentials → copy paste the code in credentials file of aws details from [details] → Save

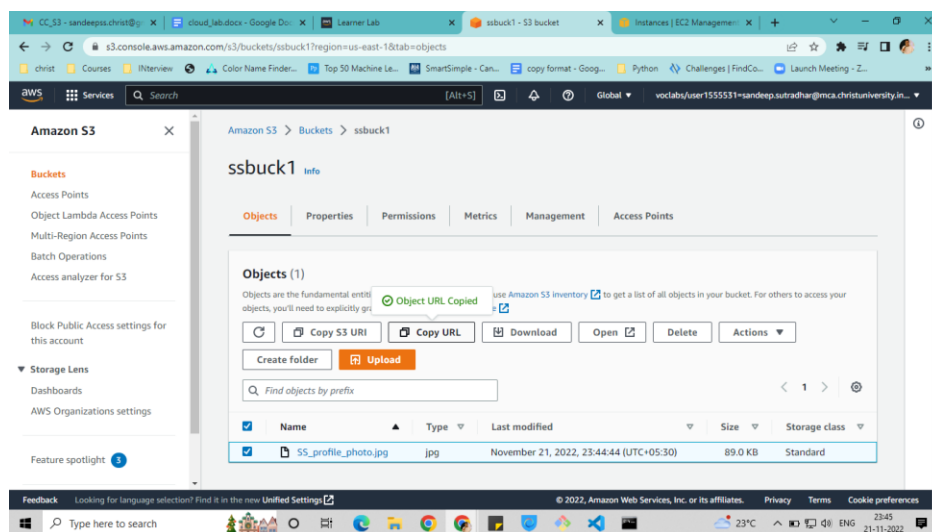
Step 3 → Open a python file (connectaws.py) → Write the following code to upload the file.

```
import boto3

s3_client = boto3.client("s3")
# response = s3_client.create_bucket(Bucket = "ssbuck1" , ACL= "public-read")
# print(response)

bucket_name = "ssbuck1"
object_name = "SS_profile_photo.jpg"
file_name = "E://christttttttttttt//trimester_5//Cloud
Computing//2147130_profile_photo.jpeg"

response = s3_client.upload_file(file_name,bucket_name,object_name)
```



Step 4 → Download the image by clicking the link https://ssbuck1.s3.amazonaws.com/SS_profile_photo.jpg

2. After the uploading of the image thumbnail images should be created for the same and it will be stored separately in a folder called “images” for the selection of the right image for the news feed.

Ans – Open another python file(ss_thumbnail.py) and write the following code –

```
import boto3
import cv2

s3 = boto3.resource("s3")
bucket_name = "ssbuck1"
orig_img_file = "E://christttttttttttt//trimester_5//Cloud
Computing//2147130_profile_photo.jpeg"
orig_image = cv2.imread(orig_img_file)

#Resizing the image to obtain thumbnail image in png format
thumbnail_image = cv2.resize(orig_image,(100,100))
thumbnail_image_png = cv2.imencode( ".png" , thumbnail_image)[1].tobytes()

#uploading the png file
```

```
thumbnail_image_filename = "thumbnail_s.png"
object = s3.Object(bucket_name, thumbnail_image_filename)
object.put(Body = thumbnail_image_png, ContentType = "image/png")
```

