Simple Storage service

- What is AWS S3?

Answer:- AWS S3 is also know as simple storage service. It is one of the storage service provided by AWS. With the help of AWS S3 we can store and access any amount of data at anytime.

- What types of data can be stored in AWS S3?

Answer:- In AWS S3 we can store any type of data like structured data like relational database , semi-structured data JSON file and unstructured data like images by using object storage using bucket.

- What are the benefits of using AWS S3?

Answer:-

- AWS s3 is highly available and durable (99.9999999999%)
- Most used storage service
- Highly scalable
- Highly economical
- Easy to manage and simple to use
- Multiple security options available like data encryption from server side and client side . And different IAM roles options.
- very flexible for different data type data storage.
- Easy to backup, disaster recovery, archive .

- How is data stored in AWS S3? (Answer with the require steps)

Answer:- data stored in AWS S3 is in object format. Object contains data , key , metadata, tag , version ID .

- To store the data in S3 we have to first create bucket and store data into it.
- Metadata consists of key value pairs and data of user and system.
- When data is added to a bucket, Amazon S3 creates a unique version ID and allocates it to the object.
- Bucket also have different IAM policies, access control list , access control points to access and manage data in AWS S3 securely .

- How do I upload data to AWS S3? (Answer with the require steps)

Answer:-

- log in to AWS console.
- Open AWS S3.
- Click on create ASE S3 Bucket .
- Write bucket name and select region and click on create bucket .
- Now create folder and give the name of the folder and select file to upload in the folder and click on upload.

- How do I make sure my data is secure in AWS S3? (Answer with the require steps)

Answer:-

- use bucket policies for give access particular user and specific access using IAM policies .

- Protect data in S3 from accidental deletion using S3 Versioning so that when ever it is required we can easily retrieve the data .
- Backup your data time to time in S3.
- Encrypt your data using server side or client side encryption in S3.
- Block public access to S3.

- How is AWS S3 priced?

Answer:- AWS S3 is charged money based on pay for what you use and billed on monthly bases.

- What are the different storage classes available in AWS S3?

Answer:- the different storage classes are divided by the data access frequency , lifecycle policies, latency , throughput and type of usage . By default all the storage classes are created as standard. AWS standard , AWS S3 Inteligent tearing, AWS S3 infrequent data access, AWS S3 one zone infrequent access, AWS Glacier.

- AWS Standard :- it is suitable for low latency and high throughput . Over here data is accessed frequently .

Example :- attendance of the student

- AWS S3 Inteligent tearing :- it is used when demand of the data or use of the data access requirement is changing according to different scenarios so it smartly automatically change to glacier to infrequent access to frequent access .
- AWS S3 infrequent data access:- it is used when data is not accessed frequently and data storage is important to store for longer duration .

Example :- MBBS student marksheet .

- AWS S3 one zone infrequent access :- it is used when data is not frequently accessed and data is stored in one region only .

Example :- student marksheet .

- AWS Glacier :- it is used when data is not used for long time and high performance is not required .

Example :- old student alumni data .

- What is the maximum size of an object that can be stored in AWS S3?

Answer:- the maximum size of an single object is 5TB and there is not any fixed number of files in the object can be stored.

- How can I enable versioning in AWS S3? (Answer with the require steps)

Answer:-

Open AWS Console :-

- Click on the Properties tab for your S3 bucket
- Click on Edit under the Bucket Versioning section
- Select Enable for Bucket Versioning
- And save changes .