

## ***AWS Test***

### **Basic-Level AWS Interview Questions:**

1. What is AWS?

Ans. AWS (Amazon Web Services) is a platform by Amazon that offers cloud computing services. Instead of buying and maintaining physical servers, companies can use AWS to rent computing power, storage, databases, and more over the internet. This saves money, improves flexibility, and allows businesses to scale up or down based on their needs.

2. What are the main services provided by AWS?

Ans. AWS has over 200 services, but the main categories are:

Compute – running applications (example EC2, Lambda),

Storage – saving files and data (example S3, EBS),

Database – managing structured and unstructured data (example RDS, DynamoDB),

Security and Identity – managing access (example IAM).

3. What is EC2?

Ans. EC2 (Elastic Compute Cloud) is a virtual machine in the cloud. You can use it like a regular computer to run websites, applications, and services. The best part is you can increase or decrease its size or number based on your app's traffic.

4. What is S3 and what are its storage classes?

Ans. S3 (Simple Storage Service) is used to store and retrieve any amount of data at any time. It's perfect for files like images, videos, and backups.

S3 offers different storage classes to match usage and cost:

Standard – for frequently accessed files,

Intelligent-Tiering – moves files between tiers based on access,

Infrequent Access – for rarely used files,

Glacier – for long-term archival storage.

5. What is the difference between EC2 and Lambda?

Ans. EC2: You manage the virtual machine yourself (like installing software and handling updates).

Lambda: You only upload your code, and AWS runs it when needed. No server setup required.

Great for small, event-driven tasks.

6. What is IAM and why is it used?

Ans. IAM (Identity and Access Management) helps you control who can access what in your AWS

account. You can create users, roles, and groups, and give them specific permissions (like “read-only access to S3”). It ensures only the right people or services can do certain actions.

7. How does AWS VPC work?

Ans. A VPC (Virtual Private Cloud) is like your own private network inside AWS. You can launch AWS resources (like EC2 instances) inside this isolated network. You get to decide the IP ranges, subnets, routing, and more – just like you would in a physical data center.

8. What is the difference between public and private subnets in a VPC?

Ans. Public subnet: Resources can connect to the internet (like web servers).

Private subnet: No direct internet access (used for secure components like databases).

9. What is an Elastic Load Balancer (ELB)?

Ans. ELB automatically distributes incoming traffic across multiple servers (EC2 instances). It ensures no single server is overloaded and helps maintain high availability of your application.

10. What is Auto Scaling in AWS?

Ans. Auto Scaling adjusts the number of EC2 instances based on traffic. If your website gets more visitors, it adds instances. When traffic drops, it removes extra ones. This helps save cost and keep the system stable.

## **Intermediate-Level AWS Questions (with explanations)**

1. What are the different types of EC2 instances?

Ans. EC2 instances come in families designed for different tasks:

T-series (General Purpose): Balanced CPU and memory.

C-series (Compute Optimized): More CPU, great for heavy processing.

M-series (Balanced): Good for most general applications.

R-series (Memory Optimized): Ideal for memory-intensive tasks like databases.

P/G-series (GPU instances): Used for AI, ML, or graphics-heavy tasks.

2. What’s the difference between EBS and S3?

Ans. EBS (Elastic Block Store) is like a hard drive attached to EC2. It stores operating systems, apps, and data you need frequently.

S3 is object storage – ideal for storing large files or media like backups, documents, or videos. It’s not attached to a specific EC2 instance.

3. How does Route 53 work in AWS?

Ans. Route 53 is a DNS service. It connects domain names (like [www.example.com](http://www.example.com)) to AWS

services or external websites. It also helps in routing users to the closest server based on location for faster performance (latency-based routing).

4. What are Security Groups and NACLs?

Ans. Security Groups: Work at the instance level. They act like firewalls that allow or block incoming and outgoing traffic.

NACLs (Network Access Control Lists): Work at the subnet level. They offer rule-based control for both inbound and outbound traffic.

5. Difference between CloudWatch and CloudTrail?

Ans. CloudWatch: Monitors resources (CPU usage, memory, etc.) and sets alerts when something goes wrong.

CloudTrail: Keeps a record of who did what in your AWS account (like who deleted an EC2 instance).

6. What is the difference between AWS RDS and DynamoDB?

Ans. RDS is a managed version of traditional relational databases (MySQL, PostgreSQL, etc.).

DynamoDB is a NoSQL database — good for fast access to unstructured or semi-structured data.

7. What is AWS Lambda's cold start issue?

Ans. When Lambda hasn't been used for a while, it takes longer to start the first time. This delay (cold start) happens because AWS needs to set up the environment. It affects performance slightly.

8. What are AWS Availability Zones and Regions?

Ans. A Region is a geographic area (like US-East or Asia-Pacific).

An Availability Zone (AZ) is a data center within that region. Using multiple AZs makes your app more reliable in case one goes down.

9. How do you secure data in transit and at rest in AWS?

Ans. In transit: Use SSL/TLS to encrypt data while it's moving (e.g., web traffic).

At rest: Use encryption options in S3, EBS, RDS, etc., or AWS Key Management Service (KMS).

10. Explain the Shared Responsibility Model of AWS.

Ans. AWS handles security of the cloud (hardware, software, network). You're responsible for security in the cloud (your apps, data, passwords, access policies). It's a partnership.

## **Advanced-Level AWS Question**

1. How would you design a fault-tolerant architecture on AWS?

Ans. To make a system fault-tolerant (i.e., continue running even if parts fail), you would:

- Use multiple Availability Zones to avoid downtime if one fails.

- Add Elastic Load Balancers (ELBs) to distribute traffic.
- Enable Auto Scaling to handle variable load.
- Store critical data in S3 or RDS Multi-AZ, with regular backups.
- Set up Route 53 with health checks for DNS-based failover.
- Monitor the system using CloudWatch and configure alerts.

This setup ensures your application remains available, even during server or zone failures.