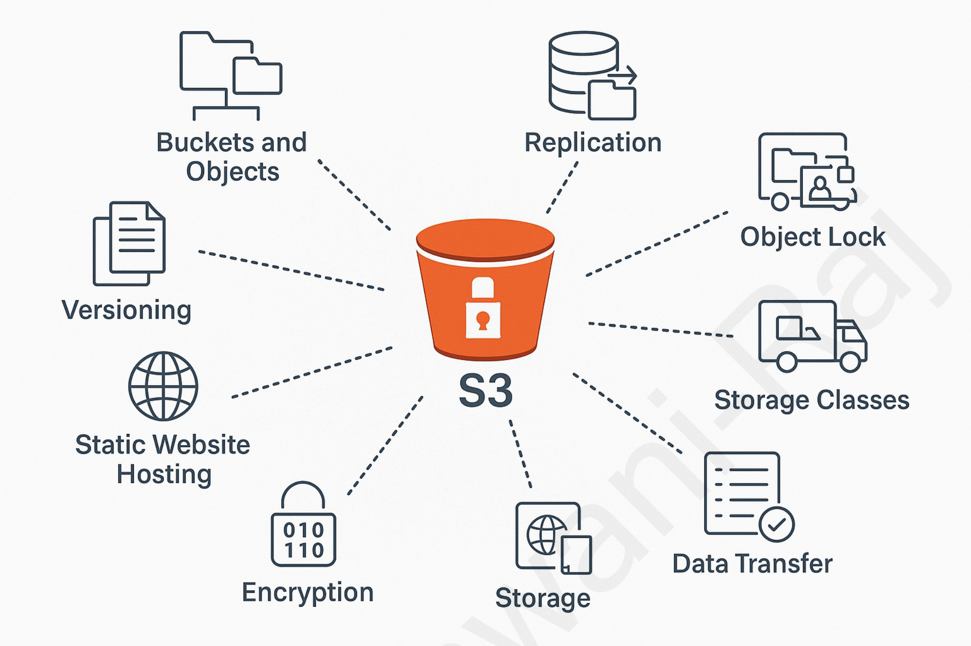
Includes

✅ **Powerful short notes on Amazon AWS Route 53**
✅ **28 frequently asked Route 53 interview questions and answers**
✅ **7 real-world scenario-based Route 53 questions**

Powerful short notes on Amazon AWS S3



1. What is AWS S3?

- AWS S3 stands for **Amazon Simple Storage Service**.
- It is a **cloud storage service** by Amazon Web Services (AWS).
- Used to **store, retrieve, backup, and share data** (like images, videos, documents, etc.) over the internet.
- Data is stored as **objects in buckets**.
- Designed for **99.999999999% (11 9's) durability**.

---

2. Key Concepts

- **Bucket**: A container for storing objects. Think of it like a folder.
- **Object**: The file/data you store in a bucket.
- **Key**: The name of the object within the bucket.
- **Value**: The content or data of the object.
- **Metadata**: Information about the object (like file type, last modified date).
- **Region**: Buckets are created in AWS regions. Choose the one closest to your users.

---

3. Steps to Create an S3 Bucket

1. Go to the AWS Management Console.
2. Navigate to **S3** under "Storage" category.
3. Click **Create bucket**.
4. Enter a **unique bucket name** (globally unique).
5. Choose a **region**.
6. Set **Bucket settings** (Block public access, Versioning, etc.).
7. Configure **tags**, **encryption**, and **permissions**.
8. Click **Create bucket**.

---

4. Uploading an Object

1. Open the bucket from the S3 console.
2. Click **Upload**.
3. Select the file(s) to upload.
4. Set **permissions** (public/private).
5. Choose **storage class**.
6. Click **Upload**.

---

5. Storage Classes

S3 offers different storage classes to save money based on access needs:

- **S3 Standard** – Frequent access.
- **S3 Intelligent-Tiering** – Automatically moves data to the best tier.
- **S3 Standard-IA (Infrequent Access)** – For less frequently used data.
- **S3 One Zone-IA** – Infrequent access in one availability zone.
- **S3 Glacier** – For archival, slow access.
- **S3 Glacier Deep Archive** – Cheapest, very slow access (hours).

---

6. Permissions and Access Control

- **Bucket Policies** – JSON policies that apply to the entire bucket.
- **IAM Policies** – Attached to IAM users, groups, or roles.
- **Access Control Lists (ACLs)** – Old method; set permissions per object.
- **Block Public Access** – Prevents accidental public access.

---

7. Encryption Options

- **SSE-S3** – AWS manages keys.
- **SSE-KMS** – You manage keys via AWS KMS.
- **SSE-C** – You provide your own encryption key.

- **Client-side encryption** – Encrypt before uploading to S3.

---

8. Versioning

- Allows multiple versions of an object in a bucket.
- Helps in recovering deleted or overwritten files.
- Enable via bucket properties.

---

9. Lifecycle Configuration

- Automate moving objects between storage classes or delete them.
- Example: Move to Glacier after 30 days.
- Reduces storage costs.

---

10. Replication

- Copy objects automatically from one bucket to another.
- **Cross-Region Replication (CRR)**: Different region.
- **Same-Region Replication (SRR)**: Same region.
- Bucket versioning must be enabled.

---

11. Static Website Hosting

- Host static websites (HTML/CSS/JS only).
- Enable from bucket properties.
- Set Index and Error documents.
- Make objects public.

---

12. Pre-Signed URLs

- Generate temporary URLs for private objects.
- You define the expiration time (e.g., 15 mins).
- Share file access securely.

---

13. S3 Events

- Trigger actions when certain events happen (e.g., object upload/delete).

- Integrated with **Lambda**, **SNS**, **SQS**.

---

14. Multipart Upload

- Upload large files in parts (required for >5 GB).
- Faster and supports retry on failed parts.

---

15. S3 Object Lock

- Prevent objects from being modified or deleted.
- Modes:
  - **Governance Mode** – Only certain users can overwrite.
  - **Compliance Mode** – No one can overwrite until retention period ends.

---

16. S3 Access Logs

- Logs all access requests to the bucket.
- Useful for audits, monitoring, and security.

---

17. Data Consistency

- S3 offers **strong consistency**:
  - Immediately read new objects after upload.
  - Immediate consistency for overwrite and delete.

---

18. Cost Considerations

You are charged for:

- Amount of storage (per GB).
- Data transfer (out of AWS).
- Requests (GET, PUT, DELETE).
- Storage class.

---

19. Tools to Use S3

- **AWS Console** – Web UI.

- **AWS CLI** – Command Line Interface.
    - Example: `aws s3 cp file.txt s3://my-bucket/`
- **SDKs** – Use with programming languages.
- **REST API** – Direct HTTP requests.

---

20. Best Practices

- Block public access unless needed.
- Enable versioning.
- Use encryption (SSE-S3 or KMS).
- Use lifecycle policies to save cost.
- Use S3 Replication for backup.
- Monitor with access logs and CloudWatch.

---

21. Common Use Cases

- Hosting static websites.
- Backup and restore.
- Data lake storage.
- Media storage and delivery.
- Disaster recovery.
- Big data analytics.

28 frequently asked Route 53 interview questions and answers

1. What is Amazon S3 and what do we mean by an S3 bucket?

Amazon S3 (Simple Storage Service) is a cloud storage service by AWS. It allows you to store and retrieve any amount of data (like files, images, videos, backups, etc.) at any time from anywhere on the internet.

An **S3 Bucket** is like a folder on the cloud where your files (called **objects**) are stored. You create a bucket first, then upload files into it. Each bucket has a unique name and is used to organize and manage your data.

---

2. What is the maximum size allowed for a single object in Amazon S3?

- The **maximum size of a single object** in S3 is **5 terabytes (TB)**.
- But if the object is **larger than 5 GB**, you must use **Multipart Upload**.
    - **Multipart Upload** splits the large file into smaller parts and uploads them in parallel, which makes the upload faster and more reliable.

---

3. How is data organized and stored in Amazon S3?

In S3, data is stored in the following way:

- **Buckets**: These are the top-level containers. You create a bucket to store files.
- **Objects**: These are the actual files/data stored inside the bucket.
- **Keys**: Each object has a key, which is like a full path or name used to identify the file in the bucket.
- **Folders**: S3 doesn't have real folders, but it shows folders in the console using slashes `/` in the object key (e.g., `images/cat.jpg` looks like a folder named "images" with a file "cat.jpg").

So, it's a flat structure, but it looks like a folder system.

---

4. What are the key differences between Amazon S3 and Amazon EBS?

Feature	Amazon S3	Amazon EBS
Type	Object storage	Block storage
Use Case	Store files, backups, images, logs	Use with EC2 for operating systems, databases
Access	Web-based access (via URL or API)	Only accessible from attached EC2 instance
Persistence	Data stored independently	Tied to EC2 instance or AZ
Performance	High for file storage	High IOPS for fast read/write
Pricing	Pay for what you store	Pay for provisioned size

In simple terms:

- **S3 is like Dropbox or Google Drive** (for files),
- **EBS is like a hard disk** connected to a virtual machine.

---

5. How do you upload a file to an S3 bucket?

You can upload files to S3 in three main ways:

1. **AWS Management Console (GUI)**
    - Go to the S3 service.
    - Choose your bucket.
    - Click “Upload” and select your file.

    - Click “Upload” again to complete.

2. **AWS CLI (Command Line Interface)**

Use a command like:

bash
CopyEdit

```
aws s3 cp myfile.txt s3://my-bucket-name/
```

3. **AWS SDKs (for programming languages like Python, Java, Node.js)**

    - Use APIs in code (e.g., using `boto3` in Python to upload a file).

---

6. Is Amazon S3 a global or regional service?

- **Amazon S3 is a regional service.**
- When you create a bucket, you choose a specific AWS region (like `us-east-1` or `ap-south-1`).
- Your data is stored in **that region**, but it can be **accessed globally** using the internet (URL or API).
- Choosing the right region helps reduce cost and improve performance.

---

7. What are the different storage classes available in Amazon S3?

S3 provides **different storage classes** based on how often you access the data and how long you keep it:

1. **S3 Standard**
    - For frequently accessed data.
    - High performance and durability.
2. **S3 Intelligent-Tiering**

    - Automatically moves data between frequent and infrequent tiers.
    - Saves money if access patterns change.
3. **S3 Standard-IA (Infrequent Access)**
    - For data not accessed often but still needed quickly.
    - Lower cost, higher retrieval cost.
4. **S3 One Zone-IA**
    - Similar to Standard-IA but stored in a single AZ.
    - Cheaper but less available.
5. **S3 Glacier**
    - For long-term archive.
    - Low cost, but retrieval takes minutes to hours.
6. **S3 Glacier Deep Archive**
    - Lowest-cost storage.
    - Designed for data rarely accessed (retrieval can take 12 hours).
7. **S3 Reduced Redundancy Storage (deprecated)**
    - Not recommended anymore.

---

8. What is versioning in S3 and why is it useful?

Versioning means keeping **multiple versions** of the same file (object) in a bucket.

- When versioning is **enabled** on a bucket:
    - Every time you upload a new version of a file, the old one is **not deleted**, just stored as an older version.
- It helps in:
    - **Recovering deleted or overwritten files.**

- **Protecting against accidental changes.**
- **Maintaining a history** of your data.

You can enable versioning from the bucket properties in the AWS Console.

---

9. How can you secure the data stored in an S3 bucket?

You can secure your S3 data using:

1. **Encryption**
   - Server-side encryption (SSE-S3, SSE-KMS, SSE-C).
   - Client-side encryption before uploading.
2. **Bucket Policies**
   - JSON-based rules applied to the bucket.
   - Define who can access the bucket and what they can do.
3. **IAM (Identity and Access Management)**
   - Control access at the user/role level using IAM policies.
4. **ACLs (Access Control Lists)**
   - Used for setting permissions at the object or bucket level.
   - Not recommended for fine-grained control.
5. **Block Public Access Settings**
   - Prevents accidental public access.
6. **MFA Delete**
   - Requires MFA (Multi-Factor Authentication) to delete objects.

---

10. What is the difference between an S3 bucket policy and an IAM policy?

Feature	S3 Bucket Policy	IAM Policy
Attached to	S3 bucket	IAM user, group, or role
Scope	Controls access to a specific bucket	Controls what AWS services a user/role can access
Use Case	Allow/deny access from users, accounts, services	Manage permissions for users in AWS account
JSON Format	Yes	Yes
Cross-account Access	Supported	Can be used with trust relationships

In simple terms:

- **Bucket policy** controls who can access the bucket and objects.
- **IAM policy** controls what AWS users or roles are allowed to do.

11. How can you make an S3 bucket 100% public for everyone to access?

To make an S3 bucket fully public (for example, to host a website or share files with everyone), follow these steps:

1. **Remove Block Public Access settings**:
    - Go to the bucket in the AWS S3 console.
    - Click on "Permissions" tab.
    - Under "Block Public Access", click "Edit".
    - Uncheck **“Block all public access”**.
    - Confirm the warning and save.

2. **Add a Bucket Policy**:
   - In the same "Permissions" tab, go to "Bucket Policy".

Paste a policy like this:

json
CopyEdit

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicRead",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::your-bucket-name/*"
    }
  ]
}
```

   -
   - Replace `your-bucket-name` with your actual bucket name.

Now, **anyone can access the files** using the object URL.

---

12. How do you host a static website using Amazon S3?

To host a static website (HTML, CSS, JS files) on S3:

1. **Create an S3 Bucket**:

    - Name it the same as your domain (e.g., `example.com` if you want to use a custom domain).

2. **Upload your website files**:
    - Upload files like `index.html`, `style.css`, etc.

3. **Make the bucket public**:
    - Disable "Block Public Access" and set a bucket policy that allows public read access.

4. **Enable static website hosting**:
    - Go to the "Properties" tab of the bucket.
    - Scroll to **"Static Website Hosting"**, click **Edit**, and enable it.
    - Set:
        - **Index document** (e.g., `index.html`)
        - **Error document** (optional, e.g., `error.html`)

5. **Use the website endpoint URL** provided by S3 to open the site.

---

13. What is an S3 Lifecycle Policy and how is it used?

An **S3 Lifecycle Policy** helps you **automatically manage the storage of your objects** over time.

You can create rules to:

- **Transition objects** to cheaper storage classes (e.g., move from Standard to Glacier after 30 days).
- **Expire objects** (e.g., delete files after 90 days).
- **Clean up old versions** if versioning is enabled.
- **Delete incomplete multipart uploads**.

This helps reduce storage costs and keeps the bucket clean.

Example use: Automatically move log files older than 60 days to Glacier.

---

14. What is multipart upload in S3 and when should you use it?

Multipart Upload is a way to upload **large files** to S3 by splitting them into smaller parts.

How it works:

- The file is divided into chunks (minimum 5 MB per part).
- Each part is uploaded in parallel.
- Once all parts are uploaded, S3 combines them into one file.

You should use it when:

- Uploading files **larger than 100 MB** (required for files over 5 GB).
- You want **faster and more reliable uploads** (even if a part fails, only that part needs to be re-uploaded).

---

15. What is S3 Transfer Acceleration and how does it help?

S3 Transfer Acceleration makes uploads and downloads **faster** by using **Amazon CloudFront's global edge locations**.

How it works:

- Instead of sending data directly to an S3 bucket in a region, it is first sent to the **nearest CloudFront edge location**.
- Then it's sent securely over Amazon's **high-speed network** to the destination bucket.

Use it when:

- Your users are **spread across the world**.
- You're uploading large files from **distant locations**.

Note: You must enable it on the bucket, and it costs extra.

16. What is a pre-signed URL in S3 and how is it used?

A **pre-signed URL** is a **temporary link** that allows someone to access a private object in S3.

How it works:

- You (or your app) generate a URL using AWS SDK or CLI.
- The link includes:
    - **Object location**
    - **Access permissions**
    - **Expiration time**
- Anyone with this link can download or upload (based on permissions) even if the object is private.

Use cases:

- Sharing a file privately.
- Allowing temporary uploads to S3 (e.g., from a user's browser).

17. What is event notification in S3 and how does it work?

S3 Event Notification allows S3 to **trigger actions when something happens** in the bucket.

You can configure S3 to send notifications for events like:

- Object created (upload completed)
- Object deleted
- Multipart upload completed

Targets for notifications include:

- **AWS Lambda** – to run custom code.
- **Amazon SQS** – to send messages to a queue.
- **Amazon SNS** – to send alerts.

Example: When a user uploads a file, S3 triggers a Lambda function that processes the file (like resizing an image).

---

18. What are the different types of encryption available in S3? (High-level overview)

There are two main types of encryption in S3:

1. Server-side Encryption (SSE):

S3 encrypts data after you upload it.

- **SSE-S3**:
  - Managed by S3 using its own keys.
- **SSE-KMS**:
  - Uses AWS Key Management Service.
  - More control and audit logging.
- **SSE-C**:
  - You provide your own encryption key.
  - AWS does not store the key.

2. Client-side Encryption:

- You encrypt the file **before uploading** it to S3.
- You manage keys and encryption on your own system.

---

19. What is S3 Object Lock and what are its use cases?

S3 Object Lock is a feature that **prevents objects from being deleted or modified** for a fixed time or forever.

Two modes:

1. **Governance Mode**:
   - Users with special permissions can override the lock.
2. **Compliance Mode**:
   - Even AWS administrators cannot delete or change the object until the retention period ends.

Use Cases:

- **Legal hold** or compliance requirements (e.g., financial or medical data).
- Protecting data from accidental or malicious deletion (e.g., ransomware protection).

---

20. What is S3 Intelligent-Tiering and when should it be used?

S3 Intelligent-Tiering is a smart storage class that **automatically moves your files** between two or more tiers based on access frequency.

Tiers:

- **Frequent Access** – for data accessed often.
- **Infrequent Access** – for rarely accessed data.
- **Archive & Deep Archive** – for long-term storage (optional).

Best used for:

- Data with **unpredictable access patterns**.
- When you don’t want to manually manage storage class transitions.
- Helps **save money** without affecting performance.

No impact on performance or retrieval time for frequent/infrequent tiers.

21. What is Amazon S3 Glacier and Glacier Deep Archive? How are they different?

S3 Glacier and **S3 Glacier Deep Archive** are **cold storage classes** used for **long-term, low-cost storage**.

🧊 **S3 Glacier:**

- Meant for data you access **once in a while** (like monthly).
- **Cheaper** than S3 Standard or Infrequent Access.
- Retrieval times:
  - Expedited: 1–5 minutes
  - Standard: 3–5 hours
  - Bulk: 5–12 hours

🧊 **S3 Glacier Deep Archive:**

- Meant for data you **rarely or never access** (like once a year).
- **Lowest-cost** storage class in S3.
- Retrieval times:
  - Standard: 12 hours
  - Bulk: 48 hours

Main Difference:

Feature	**Glacier**	**Glacier Deep Archive**
Cost	Low	Very Low
Retrieval Time	Minutes to hours	12–48 hours

Use Case	Monthly accessed archives	Yearly/legal archives

22. How can you restrict access to an S3 bucket so that only resources from a specific VPC can access it?

To allow **only a specific VPC** to access your S3 bucket:

✅ Step 1: Use a VPC Endpoint (Gateway type):

- Create a **VPC endpoint for S3**.
- This allows your VPC resources to connect to S3 **privately**, without the internet.

✅ Step 2: Set a Bucket Policy to allow only that VPC:

Use a bucket policy like:

json

CopyEdit

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Deny",
      "Action": "s3:*",
      "Resource": ["arn:aws:s3:::your-bucket-name",
"arn:aws:s3:::your-bucket-name/*"],
      "Condition": {
        "StringNotEquals": {
          "aws:SourceVpc": "vpc-xxxxxxxx"
        }
      }
    }
  ]
}
```

This **denies access to all users** who are **not from your VPC**.

23. If you accidentally deleted an object from S3, how can you recover it?

You can recover accidentally deleted objects **only if** some features were enabled before deletion:

✅ **1. Versioning Enabled:**

- If versioning was turned on, the deleted object becomes a **"delete marker"**.
- You can **restore the older version** by deleting the marker.

✅ **2. MFA Delete Enabled (optional):**

- Adds extra protection by requiring MFA to delete objects.

✅ **3. Backup Systems:**

- If you set up **replication** to another bucket or used **AWS Backup**, you can restore from there.

> ❗ If versioning and backups were not set, **you cannot recover** the deleted object.

24. What would you do if downloads from an S3 bucket are very slow?

To improve **slow downloads** from S3:

🚀 **1. Use S3 Transfer Acceleration:**

- Speeds up downloads by using **CloudFront edge locations**.
- Works well for **global users** far from the bucket's region.

🌎 **2. Use Amazon CloudFront (CDN):**

- Cache S3 files at **locations near users**.
- Reduces latency and speeds up downloads.

🌐 **3. Place buckets in nearby AWS regions:**

- If users are in Europe, use EU S3 buckets, not US-based ones.

🔁 **4. Use parallel or multipart downloads:**

- Download large files in parts or parallel streams.

---

25. How do you set up automatic backups or archiving for data stored in S3?

Here are some ways to **automate backups and archiving**:

🕒 **1. Lifecycle Policies:**

- Automatically move files to:
  - **Infrequent Access**, **Glacier**, or **Deep Archive** after some days.
  - **Delete old files** after a set period.

🔁 **2. S3 Replication:**

- Create a rule to **copy objects to another bucket** (in the same or different region).

💾 **3. AWS Backup:**

- Centralized service to **schedule and manage S3 backups**.

These methods ensure that your S3 data is safe and cost-optimized over time.

26. What steps can you take to improve S3 upload and download performance?

Here are ways to **boost performance** in S3:

⚙️ **1. Multipart Upload:**

- Upload large files in **parallel parts** (minimum 5 MB each).

🚀 **2. S3 Transfer Acceleration:**

- Speeds up upload/download by using **CloudFront edge locations**.

🌍 **3. Use CloudFront Caching:**

- Cache files close to users to reduce latency.

🧵 **4. Use parallel uploads/downloads:**

- Upload multiple files or parts **at the same time**.

🧠 **5. Choose the right region:**

- Place your S3 bucket **close to users or services** that access it.

27. What is the difference between SSE-S3 and SSE-KMS encryption in S3?

Both are **server-side encryption methods**, but they differ in **key management** and **security control**:

Feature	SSE-S3	SSE-KMS

Who manages keys?	AWS (S3)	You via AWS KMS
Key rotation?	Automatic	You can control it
Access control?	Basic (IAM policies)	Fine-grained with IAM + KMS policies
Audit logs?	No detailed logging	Detailed logging with AWS CloudTrail
Cost	No extra charge	Small cost for using KMS per request

Summary:

- Use **SSE-S3** for simple encryption needs.
- Use **SSE-KMS** for better **security, control, and auditing**.

---

28. Can Amazon S3 trigger a Lambda function? Give a real-life example.

Yes, **S3 can trigger Lambda functions** when an event happens (like a file upload).

🧪 **How it works:**

- Set up an **event notification** on the S3 bucket.
- Choose **event type** (e.g., "Object Created").
- Set **target as Lambda function**.

🧵 **Real-life example:**

Image Processing App:

- A user uploads a photo to S3.
- S3 triggers a **Lambda function**.

- Lambda resizes the image into small/medium/large sizes and saves them back to S3.

Other examples:

- Generating thumbnails
- Running antivirus scan
- Creating logs or alerts when files are uploaded

7 real-world scenario-based Route 53 questions

✅ Scenario 1: Website Hosted on EC2 - DNS Setup

Q1. *You hosted a website on an EC2 instance. How will you point your domain (like www.example.com) to this EC2 instance using Route 53?*

Answer:
To point the domain to the EC2 instance:

1. Create a **hosted zone** for your domain in Route 53.
2. Create an **A record** or **Alias record**:
   - If you have a static IP (Elastic IP) → use **A record** with the IP.
   - If you are using **Application Load Balancer** → use **Alias record** with the ALB DNS name.
3. Update your domain registrar (like GoDaddy) with the **Route 53 name servers (NS records)**.

✅ Scenario 2: High Availability Website

Q2. *You have two web servers in different regions. You want users to be routed to the nearest server for better performance. How can you do this in Route 53?*

Answer:
Use **Latency-based routing**:

1. Deploy your app in two AWS regions (e.g., US-East and Asia-Pacific).
2. Use **Route 53 latency-based routing policy** to create records for each region.
3. Route 53 will check latency and send traffic to the **closest region** with **lowest delay**.

✅ Scenario 3: Failover Setup

Q3. *Your primary website server goes down. You want traffic to automatically go to a backup server. How can Route 53 handle this?*

Answer:
Use **Failover routing policy**:

1. Set the **primary record** (pointing to main server) and attach a **health check**.
2. Set a **secondary record** (pointing to backup server).
3. If the primary server fails health check, Route 53 will automatically send traffic to the backup server.

✅ Scenario 4: Blue-Green Deployment

Q4. *You are doing blue-green deployment with two environments. How can Route 53 help you switch traffic from old version to new version?*

Answer:
Use **Weighted routing**:

1. Set two records for the same domain:
    - One for the blue environment.
    - One for the green environment.
2. Assign weights (e.g., 100 to blue, 0 to green).
3. Gradually shift traffic by adjusting weights (e.g., 80-20, 50-50, etc.).
4. When green is stable, set it to 100 and blue to 0.

✅ Scenario 5: Route Traffic Based on Country

Q5. *You want users from India to be routed to one server and users from the US to another. How can you do this in Route 53?*

Answer:
Use **Geolocation routing policy**:

1. Create a record for each location (India, US, default).
2. Route 53 will check the user's IP and direct them to the server assigned to that country.
3. Add a **default record** for all other countries not listed.

✅ Scenario 6: Domain Redirect to Another Domain

Q6. *You want to redirect traffic from `oldsite.com` to `newsite.com`. How can Route 53 help?*

Answer:
Route 53 **does not support direct HTTP redirects**, but you can:

1. Point `oldsite.com` to an **S3 bucket configured as a website redirect**.
2. In Route 53, create an **Alias A record** that points `oldsite.com` to the S3 bucket.

✅ Scenario 7: Application Hosted on CloudFront

Q7. *Your application is behind a CloudFront distribution. How do you point your domain to it using Route 53?*

Answer:

1. Create an **Alias record** (A or AAAA) in Route 53.
2. Point it to the **CloudFront distribution's DNS name**.
3. Alias records don't incur DNS query charges and support root domains like `example.com`