Day 3 – Storage & Databases (AWS Fundamentals)

🔸 Topics Covered

- **Amazon S3 (Simple Storage Service)**
- **Amazon EBS (Elastic Block Store)**
- **Amazon EFS (Elastic File System)**
- **Amazon RDS (Relational Database Service)**
- **Amazon DynamoDB**

✅ PART 1: Amazon S3 – Simple Storage Service

📘 What is S3?

- S3 is AWS's object storage service used to store and retrieve any amount of data at any time.
- Data is stored in **Buckets** as **Objects** (files).

🔨 Use Case: Host a Static Website + Backup with EBS

🔹 Step 1: Create an S3 Bucket

1. Go to AWS Console → Services → Search for **S3**
2. Click on **Create bucket**
3. Enter:
    - **Bucket Name**: `your-unique-bucket-name` (e.g., `dit-static-web-001`)
    - **Region**: Asia Pacific (Mumbai)

4. Uncheck **Block all public access**
5. Acknowledge the warning
6. Click **Create bucket**

🔹 Step 2: Upload index.html

1. Open your bucket → Click **Upload**
2. Upload your `index.html` file
3. After upload, click on the file → Permissions → Make it public (via Object actions or permissions)

🔹 Step 3: Enable Static Website Hosting

1. In the bucket → Go to **Properties**
2. Scroll to **Static website hosting**
3. Enable it
    - **Index Document**: `index.html`
4. Save
5. You'll get a **Bucket Website Endpoint** – use this to access your site

---

🔹 Step 4: Set Bucket Policy (Avoid 403 Access Denied)

1. Go to your bucket → **Permissions** tab
2. Scroll to **Bucket Policy**
3. Add this policy (replace your bucket name):

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

4. Click **Save changes**

✅ Now your HTML site is hosted via S3 static website hosting.

✅ PART 2: Amazon EBS – Elastic Block Store

📘 What is EBS?

- EBS provides **block-level storage volumes** for use with EC2 instances.
- Acts like a hard drive.

🔹 Practical: Create and Attach EBS Volume

1. Go to EC2 Dashboard → **Elastic Block Store** → Volumes
2. Click **Create Volume**
    - Type: General Purpose (gp2)
    - Size: 8 GiB
    - Availability Zone: Match with your EC2 instance (e.g., `ap-south-1a`)
3. After creating, select it → **Actions** → **Attach Volume**
    - Choose your EC2 instance

🔹 Connect via SSH to EC2

Use Git Bash or Windows Terminal to SSH

```
ssh -i your-key.pem ec2-user@your-public-ip
```

1.

Format the volume:

```
sudo mkfs -t ext4 /dev/xvdf
```

2.

Mount the volume:

```
sudo mkdir /mnt/ebs
sudo mount /dev/xvdf /mnt/ebs
```

3.

✅ Now you have persistent storage mounted to EC2.

✅ PART 3: Amazon EFS – Elastic File System

📘 What is EFS?

- EFS is a **shared file storage system** used with multiple EC2 instances.
- Ideal for applications needing shared access (e.g., web servers).

🔹 Setup EFS (Only Outline for Now):

1. Go to **EFS Service**
2. Click **Create File System**
3. Follow the wizard → Use default VPC
4. After creation, mount it using EC2 instructions provided in console

✅ Skip if you're only exploring basics.

✅ PART 4: Amazon RDS – Relational Database Service

📘 What is RDS?

- RDS is a **managed relational database** (MySQL, PostgreSQL, etc.)
- AWS handles backups, updates, security, etc.

🔹 Create MySQL Database:

1. Go to **RDS** → Click **Create database**
2. Choose:
    - Engine: MySQL
    - Template: Free Tier
3. Settings:
    - DB Instance ID: `studentdb`
    - Master username/password
4. Connectivity:
    - VPC: Default
    - Public access: **Yes**
5. Click **Create database**

⏳ Wait until it says **Available**

🔹 Connect to RDS

Use MySQL Workbench or command line:

```
mysql -h your-endpoint -u admin -p
```

1.

2. Create tables, insert data, etc.

✅ You now have a working MySQL database in RDS.

✅ PART 5: Amazon DynamoDB – NoSQL (Key-Value)

📘 What is DynamoDB?

- Fully managed NoSQL database
- Super-fast reads/writes
- No need to manage servers

🔹 Create a Table

1. Go to **DynamoDB** → Click **Create Table**
2. Table Name: `StudentData`
3. Partition Key: `StudentID` (String)
4. Keep defaults → Click **Create Table**

🔹 Add Items

1. Go to your table → Click **Explore Table Items**
2. Click **Create Item**

Enter values:

```
{
  "StudentID": "001",
  "Name": "Tino",
  "Course": "AWS"
```

```
}
```

3.
4. Save

✅ Now you've used a NoSQL store to hold student data.

🔚 Conclusion

Service	Use
S3	Store static files (website/images)
EBS	Attach storage to EC2
EFS	Shared filesystem for EC2
RDS	Managed MySQL/PostgreSQL database
DynamoDB	Key-Value NoSQL for fast access