**1. Amazon S3 (Simple Storage Service)**

**• Use Case:**

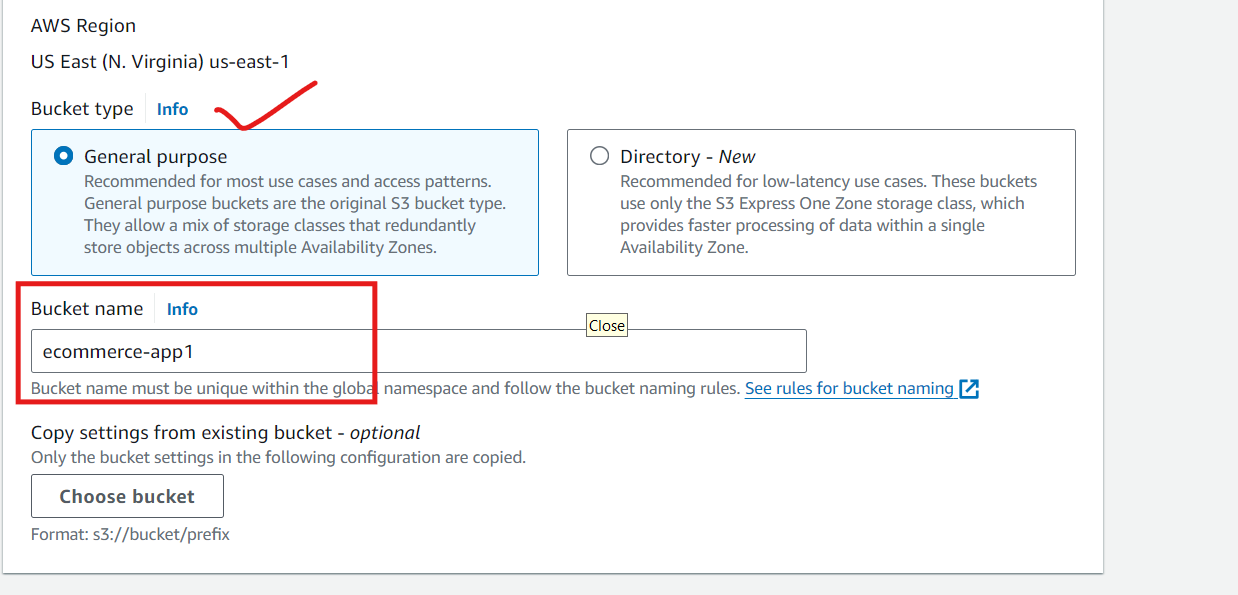
* Object storage service for storing and retrieving any amount of data,
* ideal for backups, archives, big data analytics, and web hosting.

**• Features:**

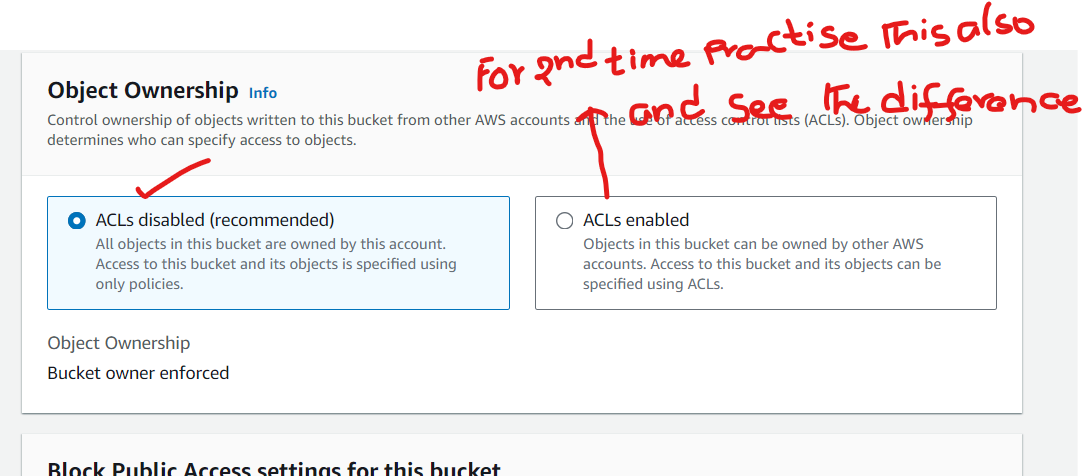
* Scalable and highly durable (11 nines of durability).
* Supports versioning, lifecycle policies, and cross-region replication.
* Offers different storage classes (e.g., S3 Standard, S3 Intelligent-Tiering, S3 Glacier).

**Creating an S3 Bucket Using the AWS Management Console**

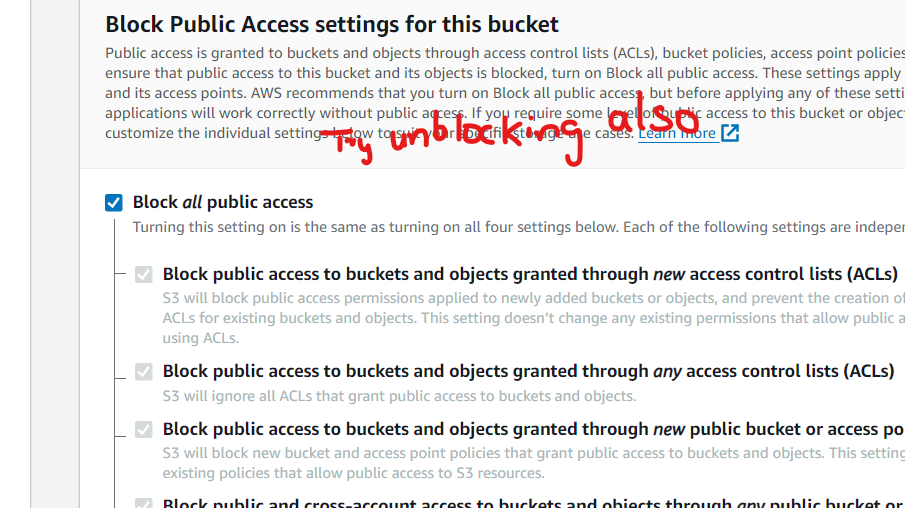
1. **Login to AWS Management Console**:
   * Go to the [AWS Management Console](https://aws.amazon.com/console/).
   * Sign in with your AWS credentials.
2. **Navigate to S3**:
   * In the "Services" menu, search for "S3" and select it.
3. **Create a New Bucket**:
   * Click on the "Create bucket" button.
4. **Configure Bucket Settings**:
   * **Bucket Name**: Enter **a unique name for** your bucket (bucket names must be globally unique across all AWS accounts).



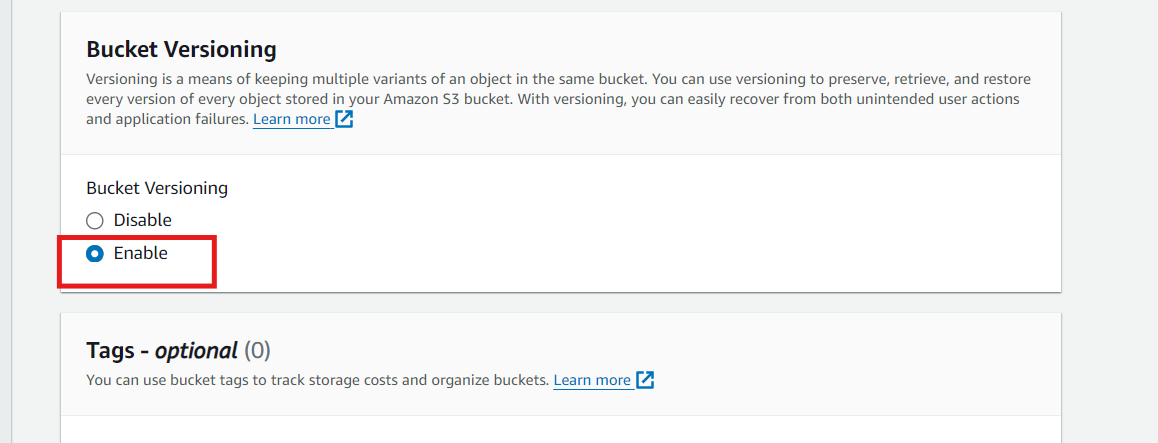
* + **Region**: Choose the AWS region where you want the bucket to reside. Selecting a region close to your users or applications can improve performance and reduce latency.
  + **Object Ownership**: Choose whether the bucket owner will control all objects (recommended for new buckets) or if it allows objects to be owned by other AWS accounts.



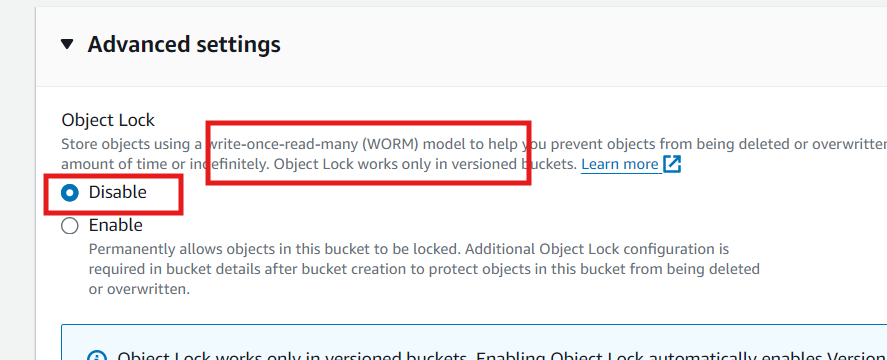
1. **Block Public Access Settings**:
   * You can choose to block all public access (recommended for most use cases). If you want your bucket to be publicly accessible, you can modify these settings accordingly.



1. **Bucket Versioning (Optional)**:
   * Enable versioning if you want to keep multiple versions of objects in the bucket. This can be useful for data protection and recovery.



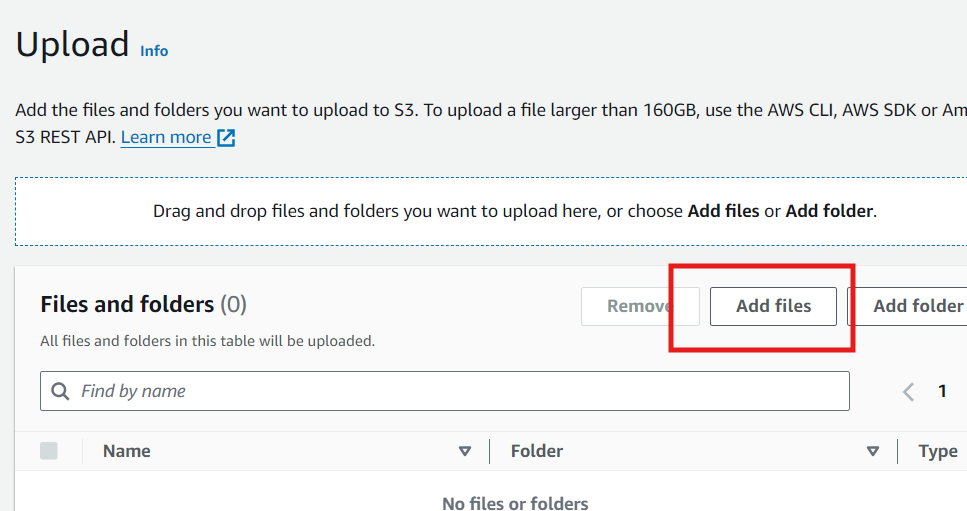
1. **Tags (Optional)**:
   * Add tags to categorize and organize your bucket resources.
2. **Default Encryption (Optional)**:
   * Enable server-side encryption to automatically encrypt objects when they are stored in S3.
3. **Advanced Settings (Optional)**:
   * You can configure additional settings like Object Lock, which helps protect objects from being deleted or overwritten.



1. **Create the Bucket**:
   * Review your settings and click "Create bucket" to finalize.

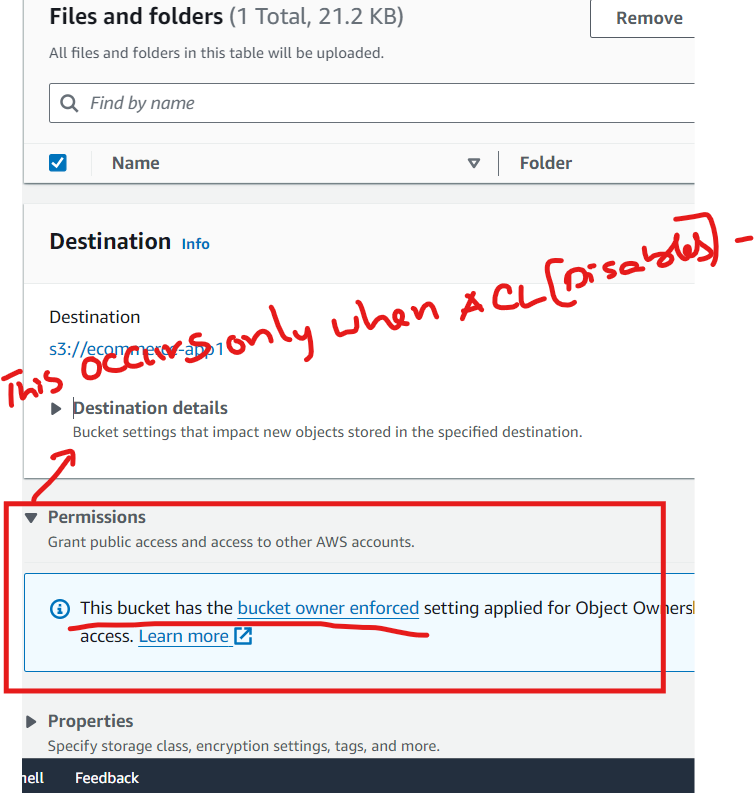
**Upload Files and Folders to the S3 Bucket**

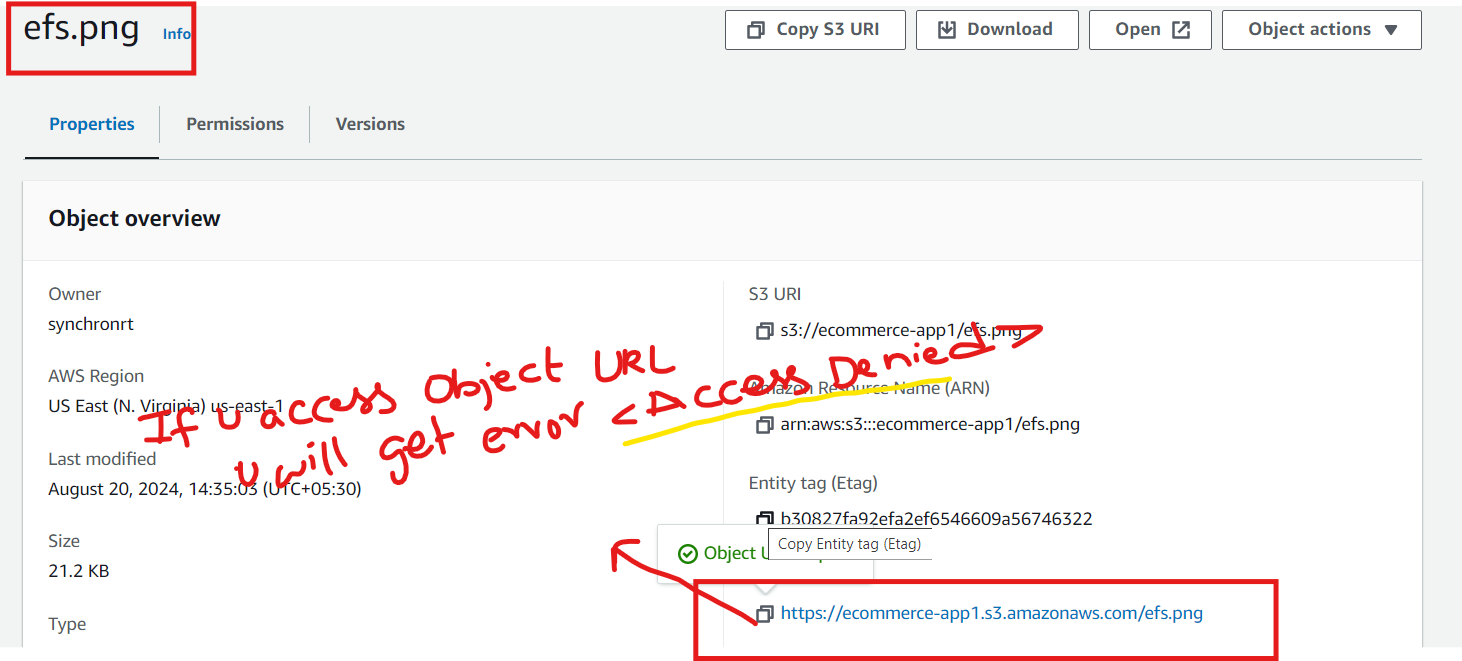
1. **Navigate to Your S3 Bucket**:
   * Go to the S3 service in the AWS Management Console.
   * Click on the bucket you created.
2. **Upload Files and Folders**:
   * Click on the “Upload” button.
   * In the upload window, you can drag and drop files or click on the “Add files” or “Add folder” button to select the files and folders from your local machine.



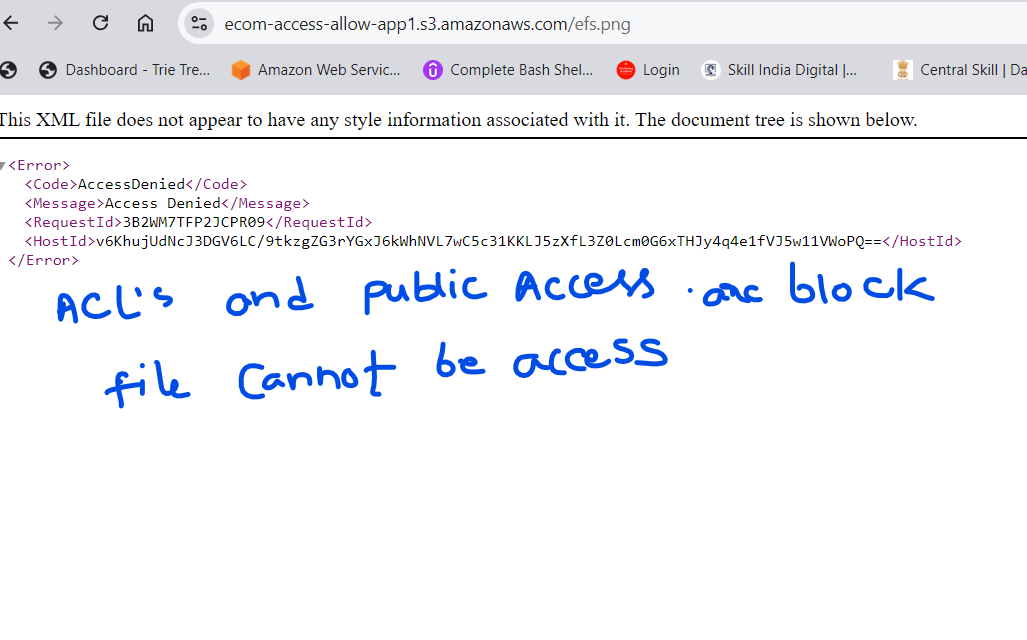
* + After selecting the files, click “Next.”

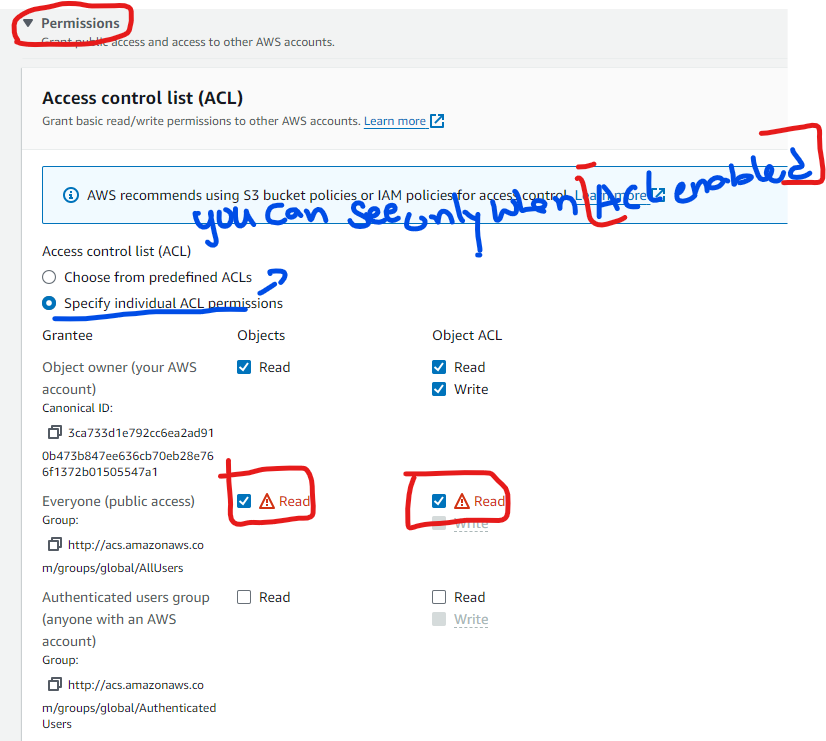
1. **Set Permissions**:
   * By default, S3 blocks all public access to the bucket. To make the files publicly accessible:
     + Click on “Manage public permissions.”
     + Select “Grant public read access to this object(s).”





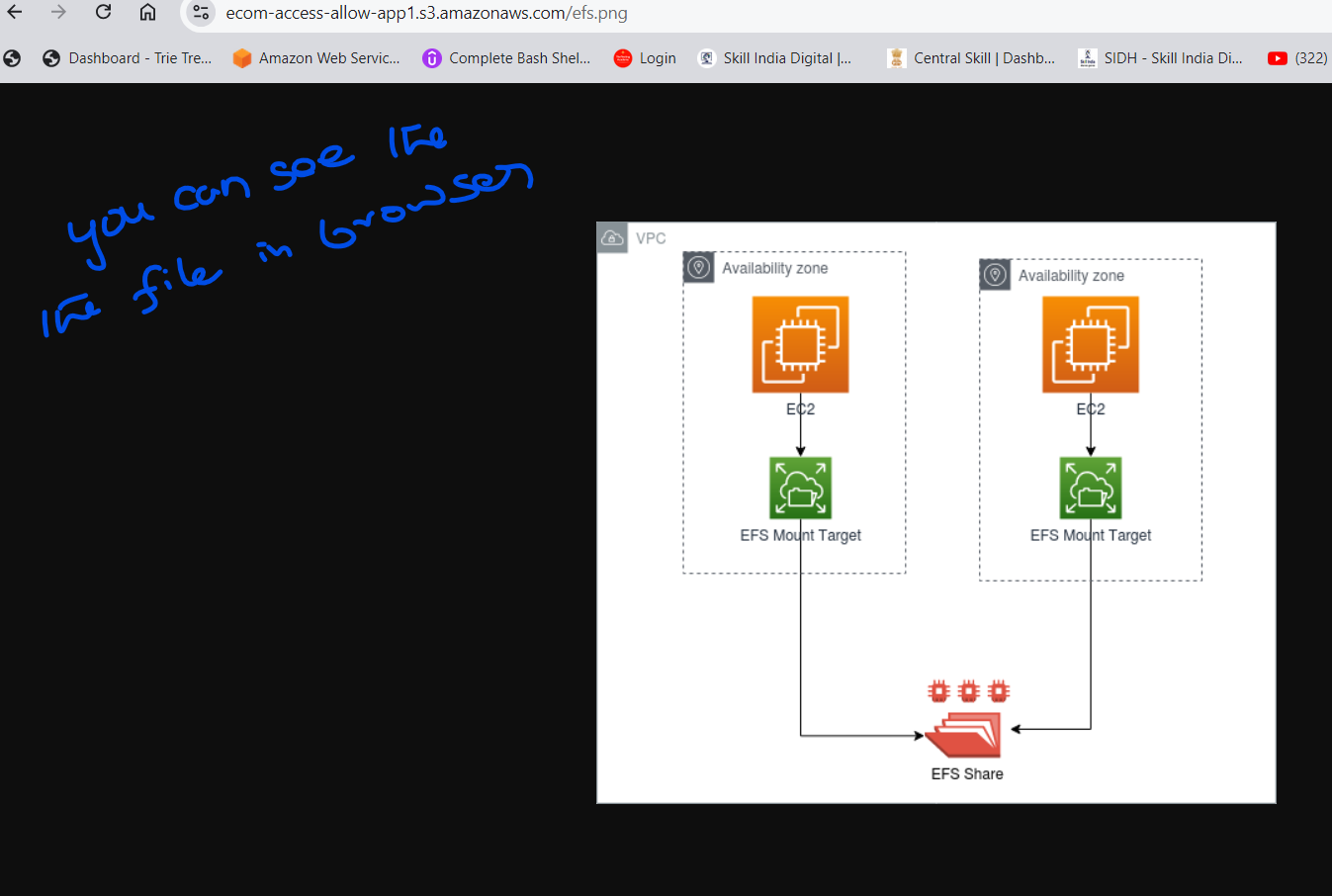
* + Click “Next” to review and confirm your settings.





1. **Storage Class**:
   * Choose the appropriate storage class based on your needs (e.g., S3 Standard for frequently accessed files).
   * Click “Next” and then “Upload” to complete the process.

**Copy the URL and hit URL in the browser**



**S3 Storage Classes Explained**

Amazon S3 offers several storage classes, each designed for different use cases and cost requirements:

1. **S3 Standard**:
   * **Use Case**: Frequently accessed data.
   * **Features**: High durability, availability, and low latency. Designed for data that requires immediate access.
   * **Cost**: The highest cost per GB but optimized for performance.
2. **S3 Intelligent-Tiering**:
   * **Use Case**: Data with unpredictable access patterns.
   * **Features**: Automatically moves data between two access tiers (frequent and infrequent) based on changing access patterns, optimizing costs.
   * **Cost**: Slightly higher than S3 Standard when frequently accessed but lower for infrequent access.
3. **S3 Standard-IA (Infrequent Access)**:
   * **Use Case**: Data that is accessed less frequently but requires rapid access when needed.
   * **Features**: Lower storage cost compared to S3 Standard but higher retrieval costs. Ideal for backups and long-term storage of less frequently accessed data.
   * **Cost**: Lower storage cost with a per-GB retrieval fee.
4. **S3 One Zone-IA**:
   * **Use Case**: Infrequently accessed data that doesn’t require the resilience of being stored across multiple availability zones.
   * **Features**: Lower cost than S3 Standard-IA but only stored in one availability zone, so it's less durable in the event of an AZ failure.
   * **Cost**: Even lower than Standard-IA but with less durability.
5. **S3 Glacier**:
   * **Use Case**: Archival data that is rarely accessed and requires flexible retrieval times, ranging from minutes to hours.
   * **Features**: Very low-cost storage designed for long-term data archiving. Retrieval can take minutes to hours depending on the retrieval option selected.
   * **Cost**: Very low storage cost, but retrieval incurs additional costs based on retrieval time (expedited, standard, or bulk).
6. **S3 Glacier Deep Archive**:
   * **Use Case**: Long-term data archiving with retrieval times in hours (typically 12 hours).
   * **Features**: The lowest-cost storage option available in S3. Retrieval times are longer, making it suitable for data that is rarely accessed and can withstand retrieval delays.
   * **Cost**: The lowest cost per GB, with higher retrieval costs and longer retrieval times.
7. **S3 Outposts**:
   * **Use Case**: Storing S3 data on-premises using AWS Outposts.
   * **Features**: For workloads that require local data storage on AWS-managed infrastructure at on-premises locations.
   * **Cost**: Similar to S3 Standard, but designed for specific on-premises use cases.

**Amazon Elastic Block Store (EBS):**

* **Use Cases:**
  + **High-performance databases:** EBS is ideal for applications that require consistent, low-latency block storage, such as relational (e.g., MySQL, PostgreSQL) and NoSQL databases (e.g., MongoDB).
  + **Transactional workloads:** Applications that need to process high I/O operations per second (IOPS), like financial transaction systems.
  + **Boot volumes for EC2 instances:** EBS can be used as the root volume for an EC2 instance, storing the operating system and application data.
  + **Data warehousing and Big Data analytics:** When dealing with structured data that needs high throughput and low latency.
* **Suitable Applications:**
  + Relational databases (e.g., MySQL, Oracle)
  + NoSQL databases (e.g., Cassandra, MongoDB)
  + Enterprise applications (e.g., SAP, Microsoft Exchange)
  + Data warehousing solutions (e.g., Amazon Redshift)

**2. Amazon Elastic File System (EFS):**

* **Use Cases:**
  + **Shared file storage:** EFS is perfect for applications that require a common file system to be accessible by multiple EC2 instances simultaneously, such as content management systems.
  + **Big Data and analytics:** EFS can handle a large volume of data and can scale as data grows, making it suitable for processing large datasets.
  + **Content management and web serving:** Applications that need to store and serve content, such as media files, across multiple servers.
  + **Container storage:** EFS can be used to persist data for containerized applications in services like Amazon ECS or Kubernetes.
* **Suitable Applications:**
  + Content management systems (e.g., WordPress, Joomla)
  + Media processing (e.g., video rendering, image processing)
  + Web server farms (e.g., hosting web applications across multiple EC2 instances)
  + DevOps tools requiring shared storage (e.g., CI/CD pipelines)

**3. Amazon Simple Storage Service (S3):**

* **Use Cases:**
  + **Backup and disaster recovery:** S3 is widely used for storing backups due to its durability and scalability.
  + **Data lakes and big data analytics:** S3 can store vast amounts of unstructured data, making it ideal for big data applications and machine learning.
  + **Content distribution:** With integration to services like CloudFront, S3 is used to distribute content globally.
  + **Static website hosting:** S3 can host static websites, serving HTML, CSS, and JavaScript files directly from the bucket.
  + **Archiving:** S3's various storage classes, including S3 Glacier, are perfect for long-term archival of data.
* **Suitable Applications:**
  + Static website hosting
  + Media storage and distribution (e.g., videos, images, backups)
  + Big data storage (e.g., data lakes, analytics)
  + Backup and archiving solutions