

## ✅ Task Details and Deliverables

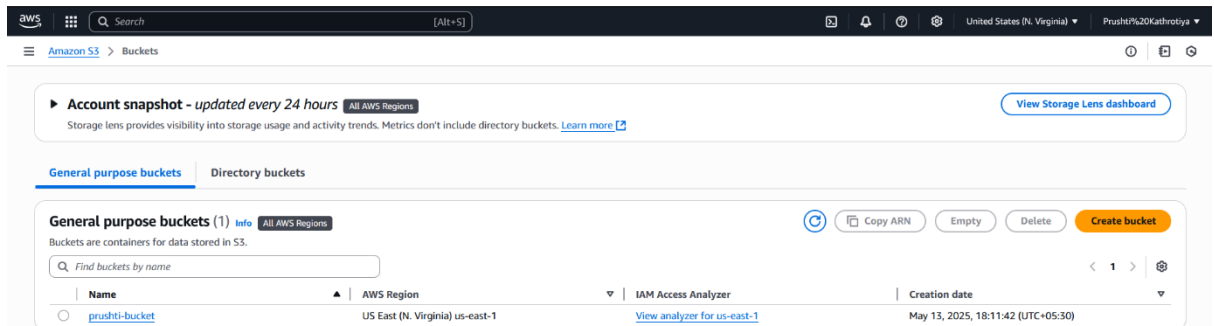
### 🚀 Task 1: Cloud Storage Setup

- **Objective:**

Create and configure a cloud storage solution using either **AWS S3** or **Google Cloud Storage**.

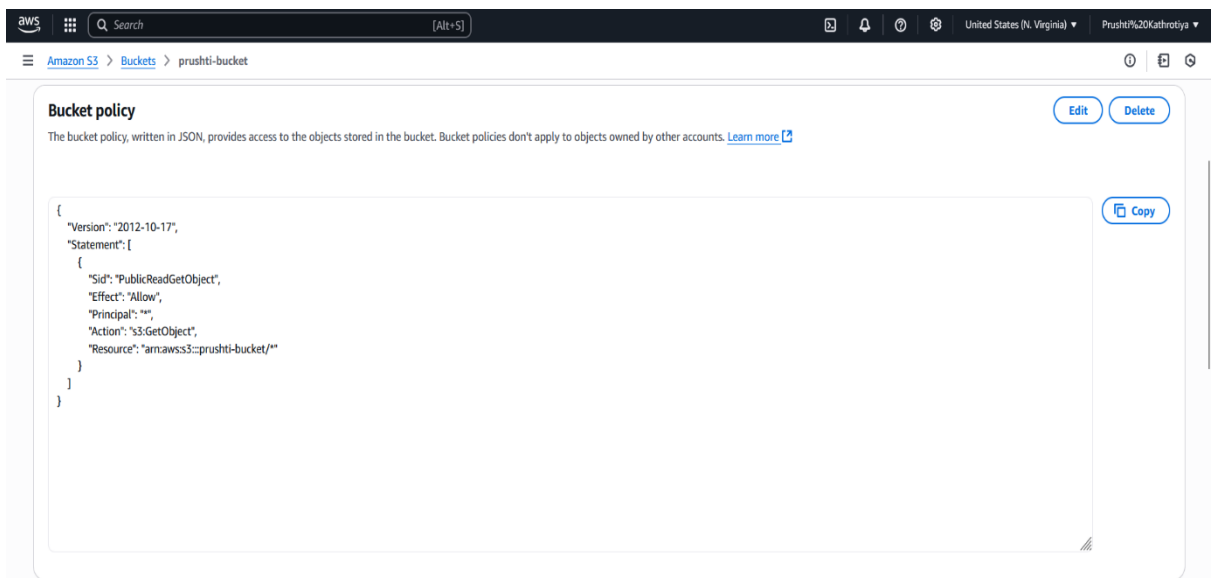
- **Instructions:**

- Create a storage bucket
- Upload sample files
- Configure access permissions (public/private)



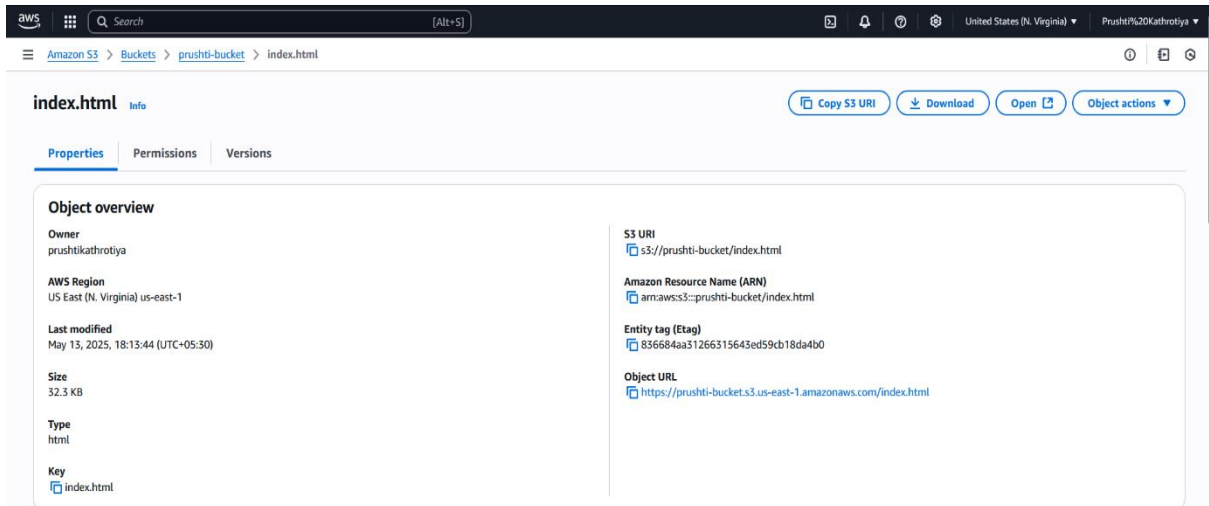
The screenshot shows the AWS Management Console 'Buckets' page. At the top, there's a navigation bar with the AWS logo, a search bar, and user information. Below the navigation bar, there's a section for 'Account snapshot - updated every 24 hours' with a 'View Storage Lens dashboard' link. The main content area is titled 'General purpose buckets (1)' and includes a 'Create bucket' button. Below this, there's a table listing the buckets. The table has columns for Name, AWS Region, IAM Access Analyzer, and Creation date. A single bucket named 'prushti-bucket' is listed, located in the 'US East (N. Virginia) us-east-1' region, with a creation date of 'May 13, 2025, 18:11:42 (UTC+05:30)'.

Name	AWS Region	IAM Access Analyzer	Creation date
<a href="#">prushti-bucket</a>	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	May 13, 2025, 18:11:42 (UTC+05:30)



The screenshot shows the 'Bucket policy' page for the 'prushti-bucket'. It displays the bucket policy in JSON format. The policy allows the 'PublicReadGetObject' action for the principal '\*' (anyone) on the resource 'arn:aws:s3:::prushti-bucket/\*'. There are 'Edit' and 'Delete' buttons at the top right, and a 'Copy' button next to the JSON code.

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



- **Deliverables:**

- Screenshots of the bucket configuration:

- **Demo Video:**

- [Click to Watch Task 1 Demo](#)

- Brief setup guide or explanation:

The bucket was created using AWS S3. Sample files were uploaded and made public by configuring bucket permissions. IAM roles were used to limit unauthorized access.