SIMPLE STORAGE SERVICE (S3)

Amazon S3 (**Simple Storage Service**) is a scalable, high-performance, and highly available **object storage** service offered by Amazon Web Services (AWS). It is designed to store and retrieve any amount of data from anywhere on the internet.

Key Features of Amazon S3

1. **Scalable Storage**:
    - Automatically scales to accommodate any amount of data.
2. **Durability and Availability**:
    - Provides **99.999999999% (11 nines)** durability and high availability across multiple AWS regions.
3. **Security**:
    - Supports data encryption at rest and in transit, along with fine-grained access controls using AWS Identity and Access Management (IAM) policies.
4. **Cost-Effective**:
    - Pay-as-you-go pricing model; various storage classes are available for different use cases.
5. **Global Accessibility**:
    - Accessible from anywhere via a web interface or API.

Common Use Cases for Amazon S3

1. **Backup and Archiving**:
    - Securely store backups and archived data.
2. **Web Hosting**:
    - Host static websites or serve static assets like images, videos, or JavaScript files.
3. **Big Data Analytics**:
    - Store large datasets for analytics or machine learning applications.

4. **Application Hosting**:
    - Store and distribute application data such as user uploads or configuration files.
5. **Media Storage**:
    - Host large media files, such as video streaming or image galleries.

How Amazon S3 Works

1. **Buckets**:
    - All data is stored in **buckets**, which are top-level containers for organizing objects.
2. **Objects**:
    - Each piece of data (e.g., a file) is stored as an **object** inside a bucket. Objects consist of data, metadata, and a unique key.
3. **Storage Classes**:
    - Different storage classes (e.g., Standard, Glacier, One Zone-IA) offer varying levels of performance, durability, and cost.
4. **Access Management**:
    - Access to S3 resources is managed using IAM policies, bucket policies, or Access Control Lists (ACLs).

S3 Storage Classes

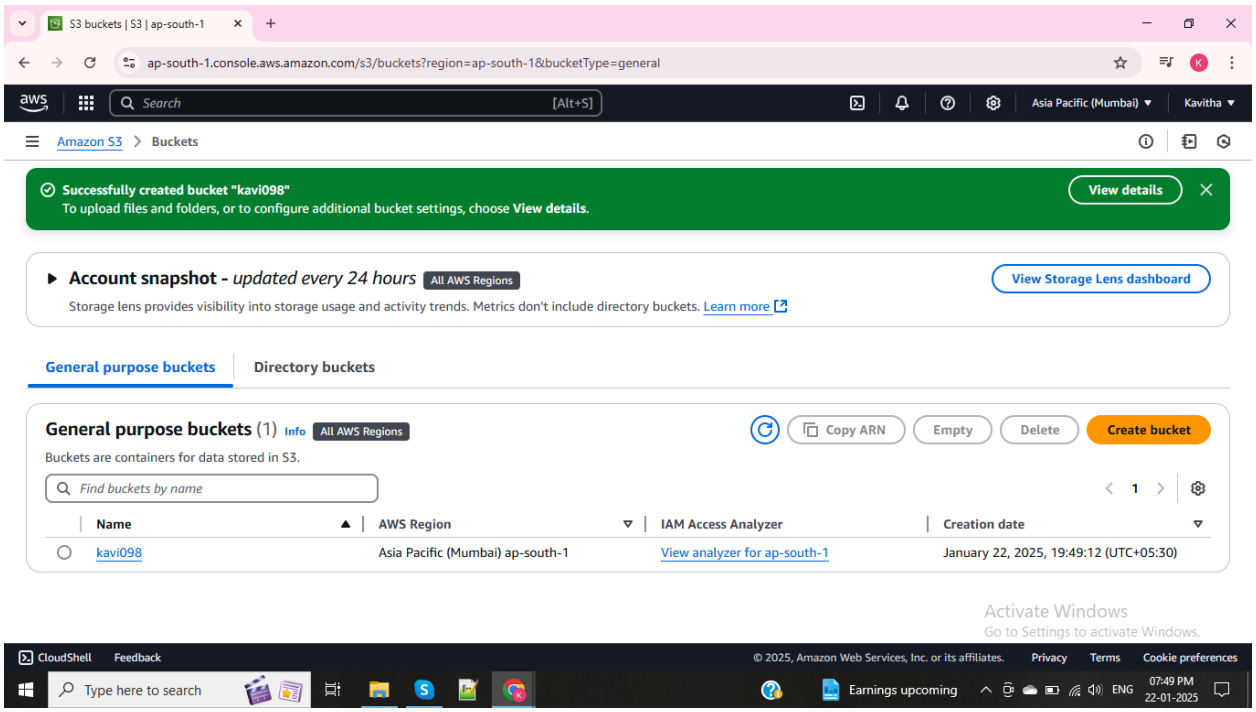
1. **S3 Standard**: General-purpose storage for frequently accessed data.
2. **S3 Intelligent-Tiering**: Automatically moves data to the most cost-effective storage tier.
3. **S3 Glacier**: Designed for long-term archival storage at the lowest cost.
4. **S3 One Zone-IA**: For infrequently accessed data stored in a single availability zone.
5. **S3 Glacier Deep Archive**: Ultra-low-cost storage for long-term archiving.

Amazon S3 is versatile and widely used across industries for its scalability, durability, and integration with other AWS services.

TASK 6 : How to Host a Static Website in Simple Storage Service (S3) in AWS

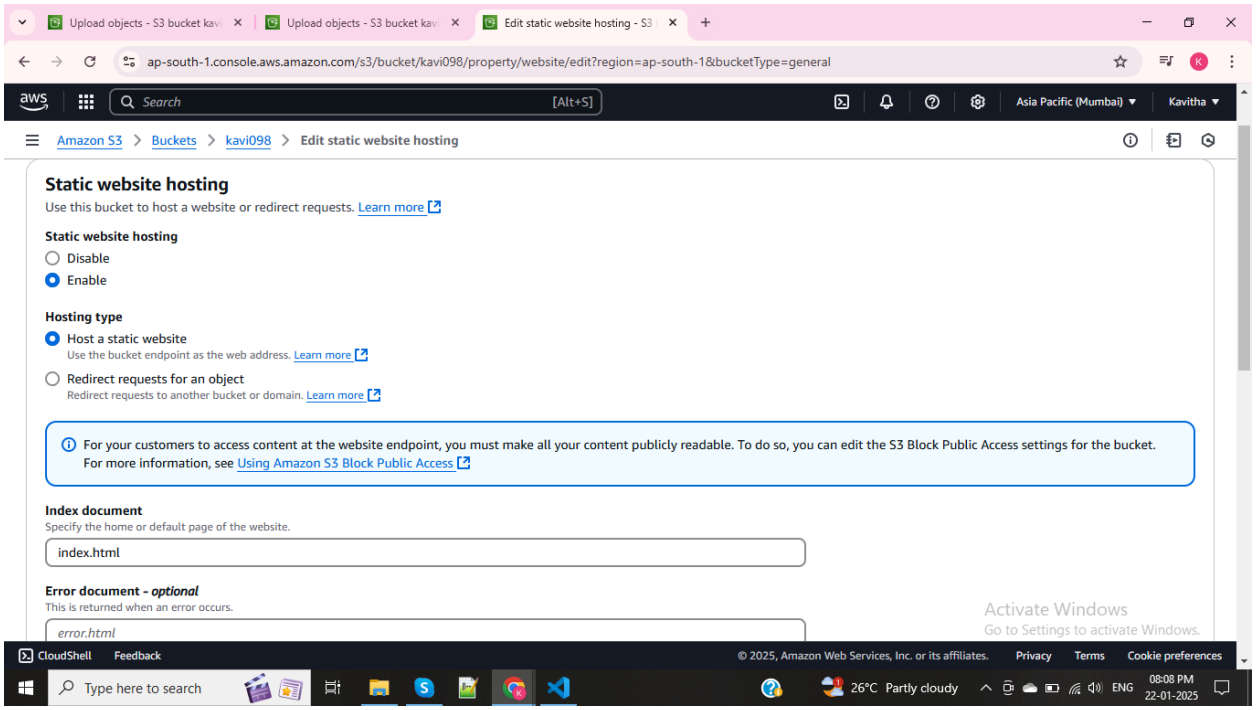
Step 1: Create an S3 Bucket

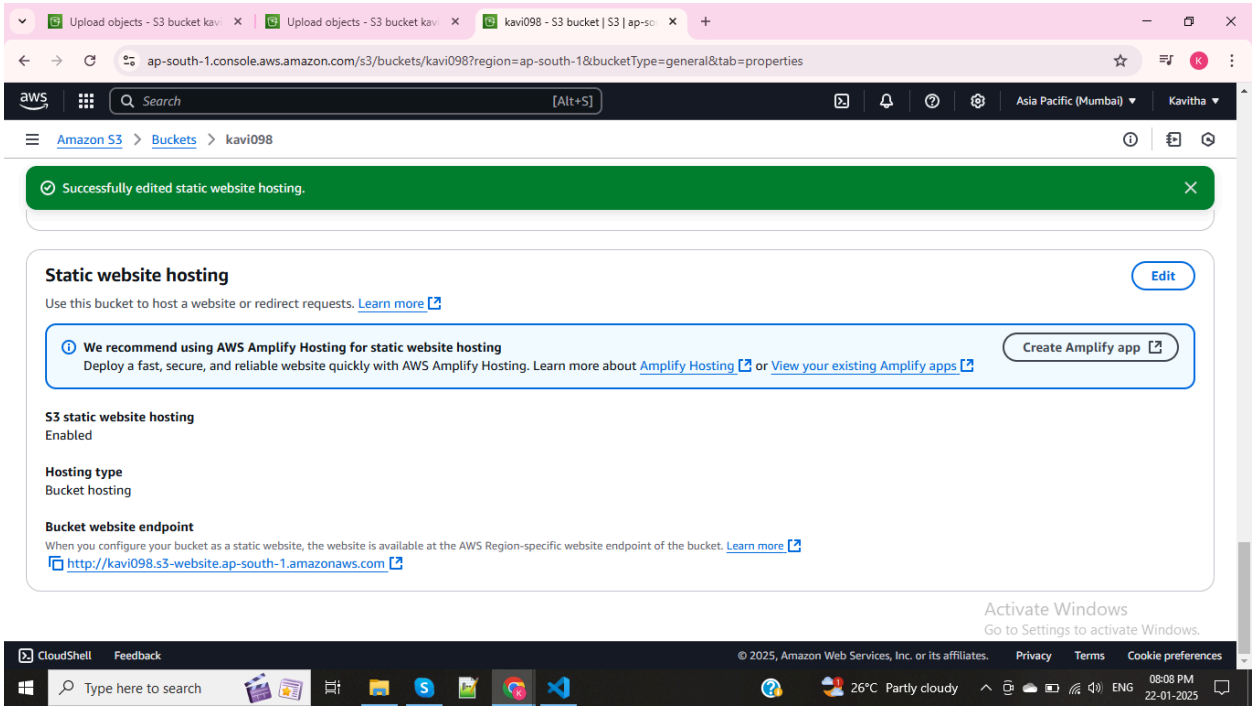
1. **Go to the AWS Management Console**:
    - Open the S3 console.
2. **Click on "Create bucket"**:
    - Enter a globally unique bucket name --> Click **Create bucket**.



Step 2: Enable Static Website Hosting

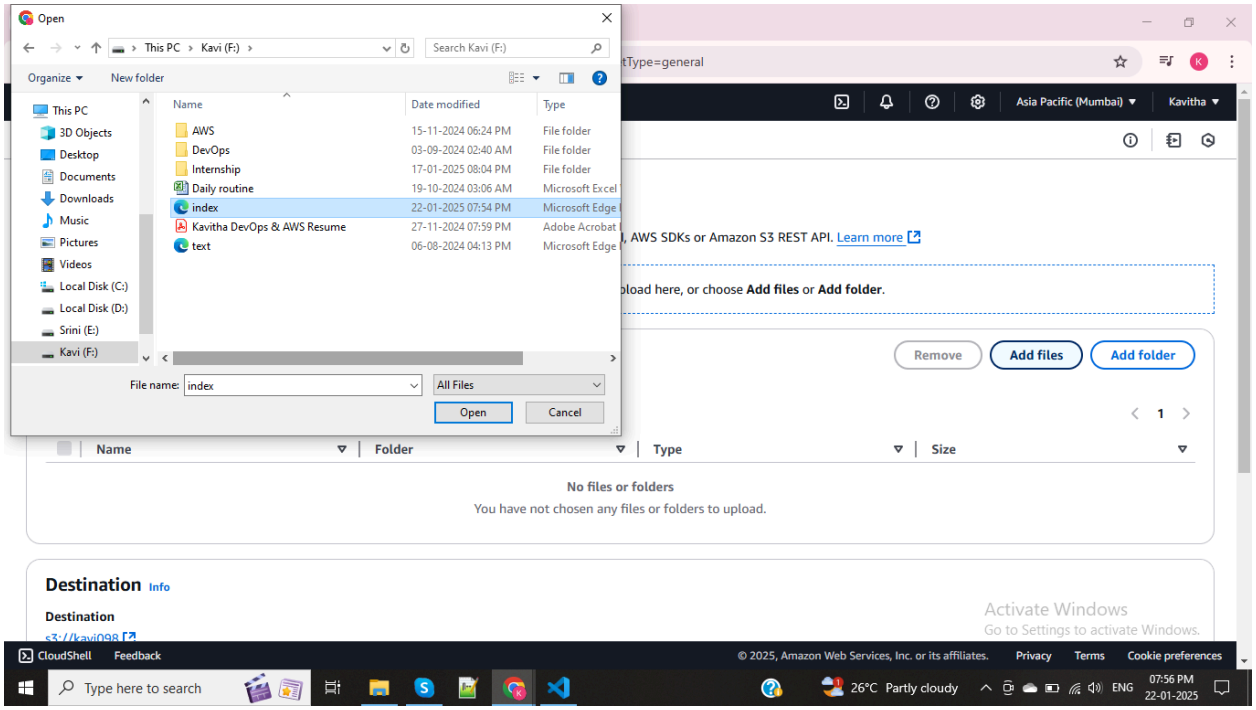
1. Select the bucket you created.
2. Go to the **Properties** tab.
3. Scroll down to **Static website hosting** and click **Edit**.
4. Enable **Static website hosting**.
5. Specify the **index document** (`index.html`)
6. Save the changes.

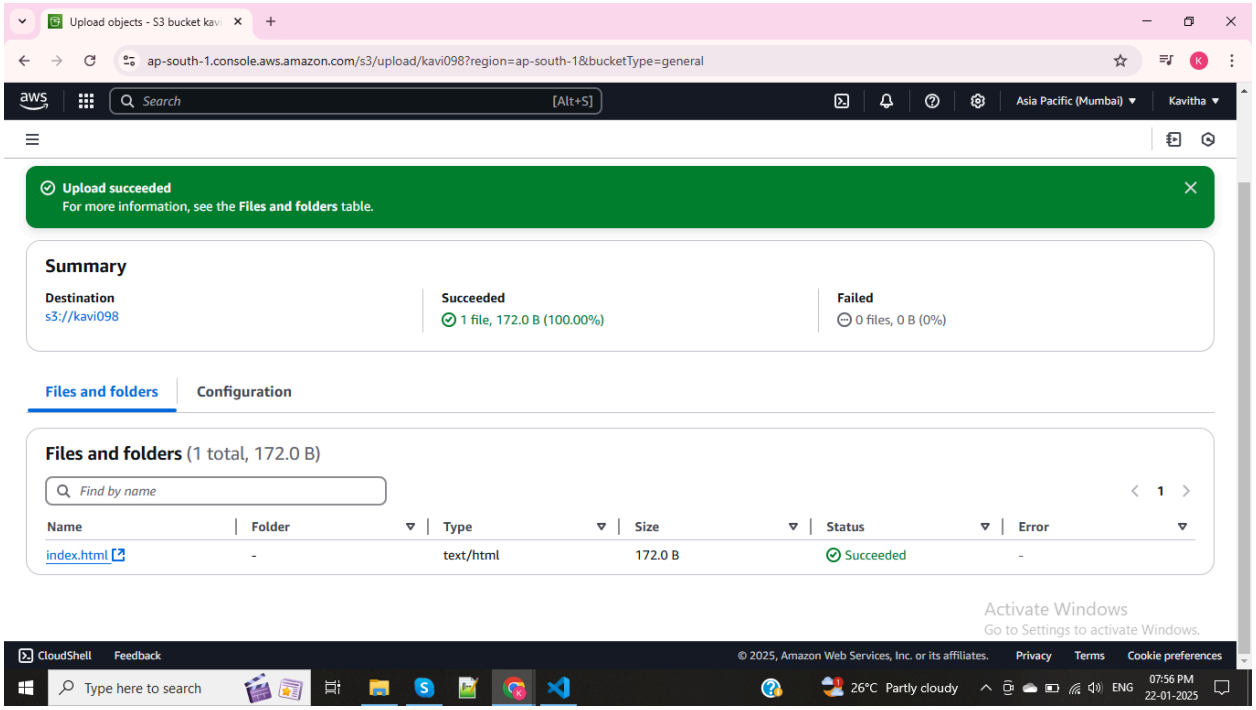




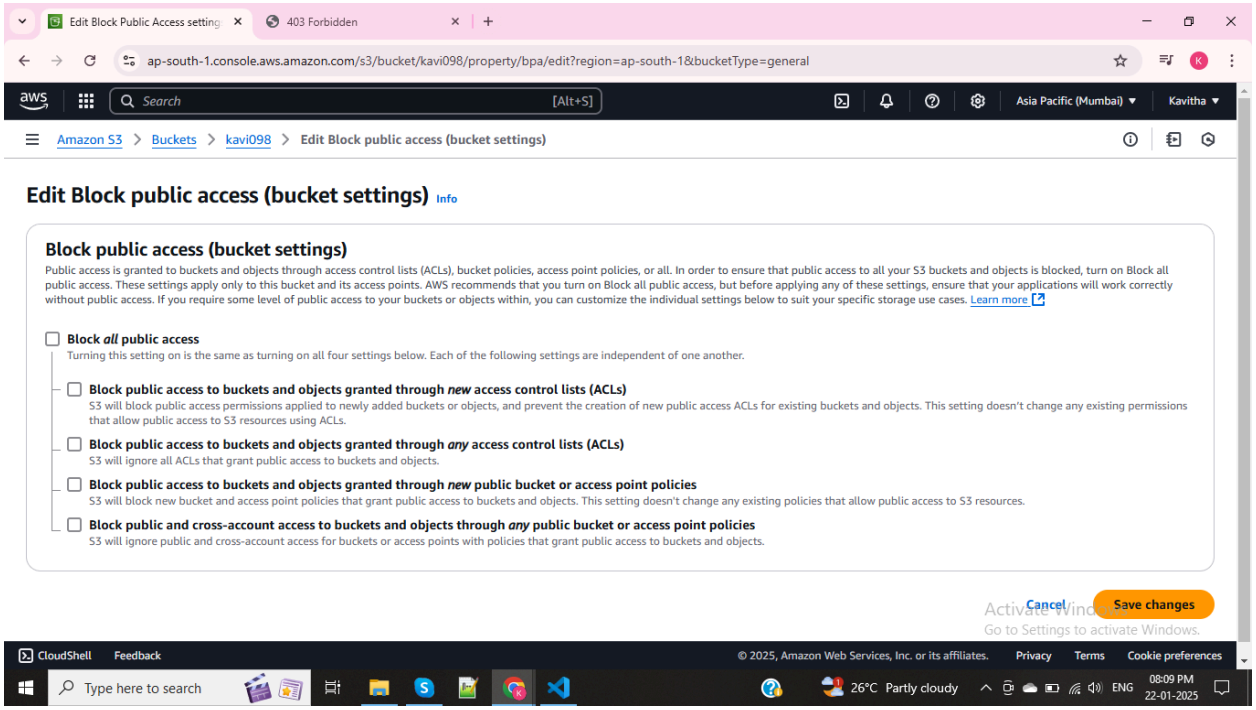
Step 3: Upload Your Website Files

1. Go to the **Objects** tab in the bucket.
2. Click **Upload**.
3. Upload your HTML file.


Open
This PC › Kavi (F:) ›
Search Kavi (F:)
Organize
New folder
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (D:)
Srini (E:)
Kavi (F:)
Name
Date modified
Type
AWS 15-11-2024 06:24 PM File folder
DevOps 03-09-2024 02:40 AM File folder
Internship 17-01-2025 08:04 PM File folder
Daily routine 19-10-2024 03:06 AM Microsoft Excel
index 22-01-2025 07:54 PM Microsoft Edge
Kavitha DevOps & AWS Resume 27-11-2024 07:59 PM Adobe Acrobat
text 06-08-2024 04:13 PM Microsoft Edge
File name: index
All Files
Open
Cancel
Asia Pacific (Mumbai)
Kavitha
AWS SDKs or Amazon S3 REST API. Learn more
load here, or choose Add files or Add folder.
Remove
Add files
Add folder
Name
Folder
Type
Size
No files or folders
You have not chosen any files or folders to upload.
Destination Info
Destination
Activate Windows
Go to Settings to activate Windows.
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
26°C Partly cloudy
ENG
07:56 PM
22-01-2025

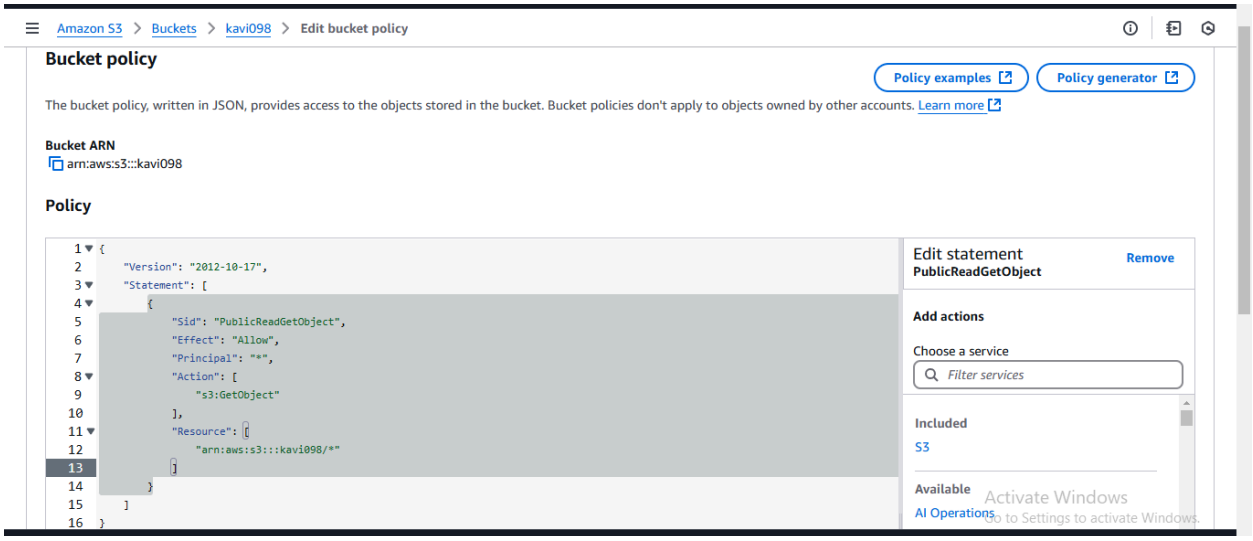

Upload objects - S3 bucket kavi
ap-south-1.console.aws.amazon.com/s3/upload/kavi098?region=ap-south-1&bucketType=general
Search
[Alt+S]
Asia Pacific (Mumbai)
Kavitha
Upload succeeded
For more information, see the Files and folders table.
Summary
Destination
s3://kavi098
Succeeded
1 file, 172.0 B (100.00%)
Failed
0 files, 0 B (0%)
Files and folders
Configuration
Files and folders (1 total, 172.0 B)
Find by name
Name
Folder
Type
Size
Status
Error
index.html
-
text/html
172.0 B
Succeeded
-
Activate Windows
Go to Settings to activate Windows.
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
26°C Partly cloudy
ENG
07:56 PM
22-01-2025

Uncheck the "Block all public access" option to access the website



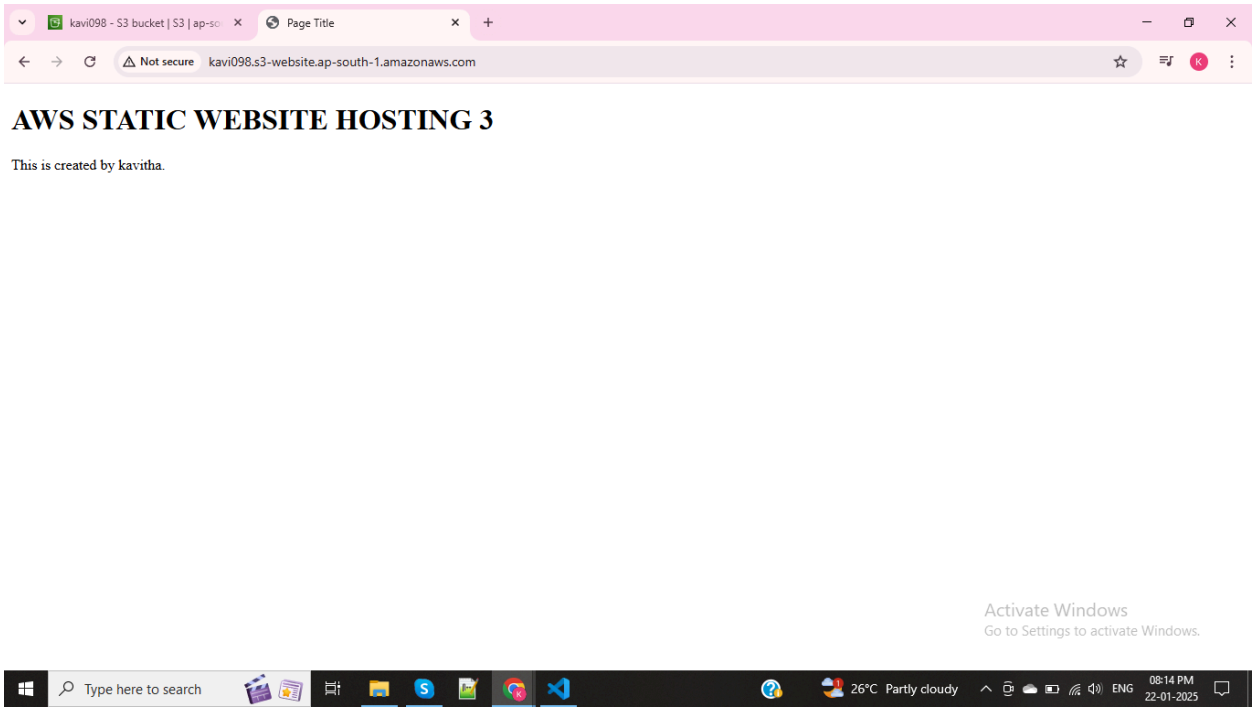
Step 4: Configure Permissions

1. **Update Bucket Policy**:
    - Go to the **Permissions** tab.
    - Scroll to **Bucket policy** and click **Edit**.
    - Add the following policy to make the bucket's contents publicly accessible



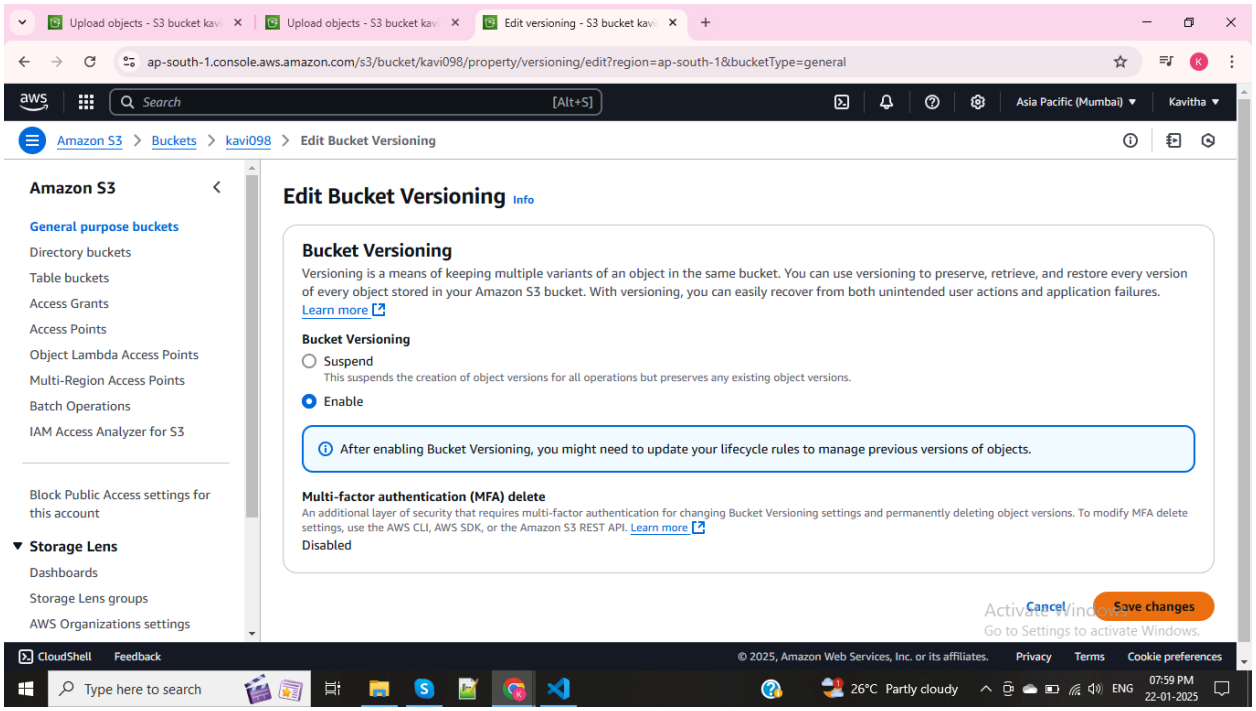
Step 5: Test the Website

1. Go back to the **Properties** tab.
2. Under **Static website hosting**, copy the **Endpoint URL**.
3. Open the URL in a browser to see your website.

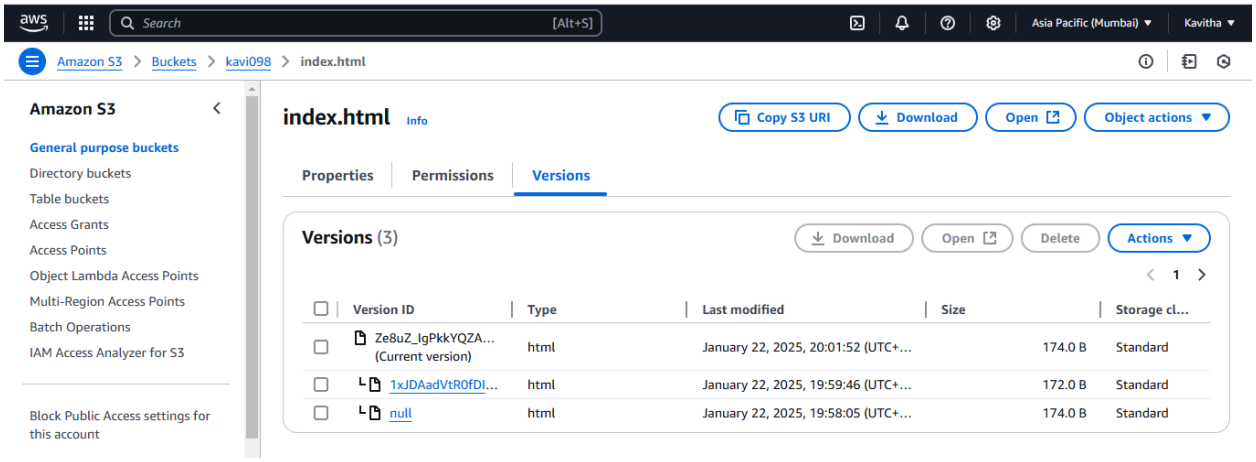


Bucket versioning in Amazon S3 is a feature that enables you to keep multiple versions of an object in a bucket. With versioning enabled, every time you overwrite or delete an object, S3 saves the previous version(s) instead of permanently replacing or removing it.

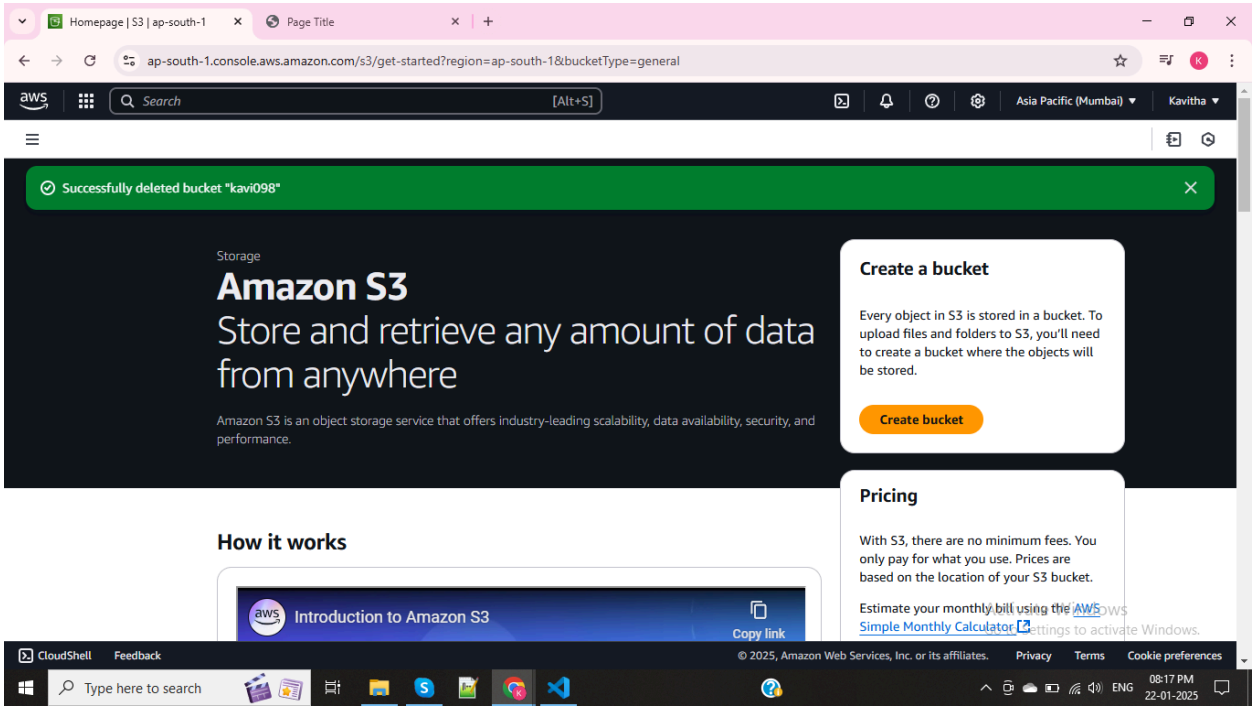
Go to the **S3 Service** --> Select the bucket where you want to enable versioning --> Navigate to the **Properties** tab --> Under the **Bucket Versioning** section, click **Edit** --> Choose **Enable** and save the changes.



Uploading a File --> Updating the File and save



For Deleting the Bucket first we have to empty the bucket and then it access to delete.



Index.html file