

**Name: Prasad Deshpande**

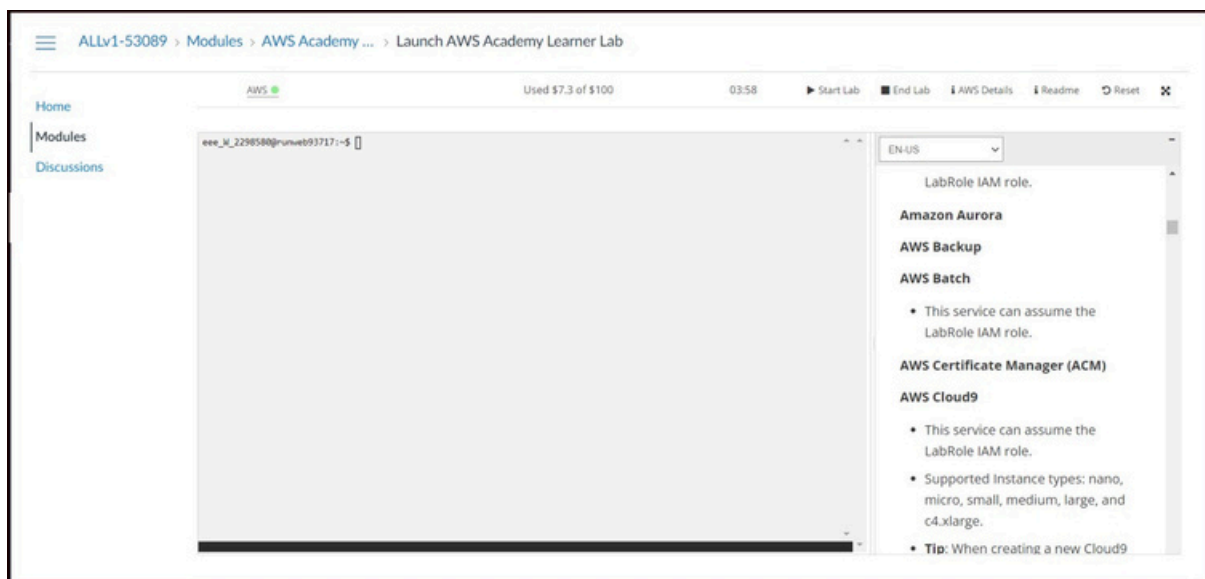
**Enrollment Number: 243341024**

**MSC(CS) Part I**

## **Cloud Computing Practical Assignment No 8**

Hosting a static website in AWS using S3

First of all open Virtual Lab. Then click on the Start Lab button. When the circle icon to the right of the AWS link in the upper-left corner turns green, it indicates that the lab environment is ready to use this we can see in Fig 1. To launch the AWS Management Console in a new tab, select the AWS link



**Fig 1**

After clicking on "AWS" we get the console home as shown in Fig 2. Then click on "S3" service.

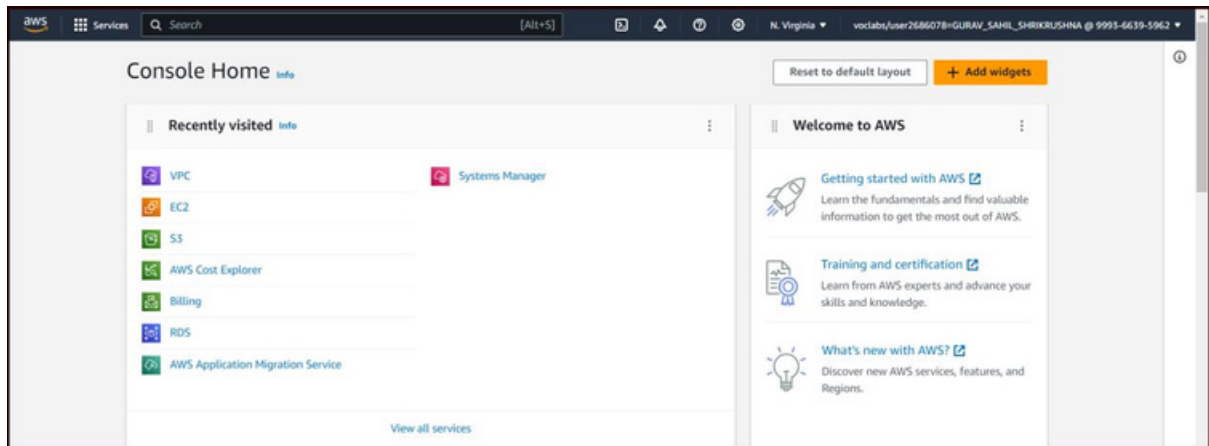


Fig 2

In Fig 3 we can see the Amazon S3 interface in which click “Create Bucket”.

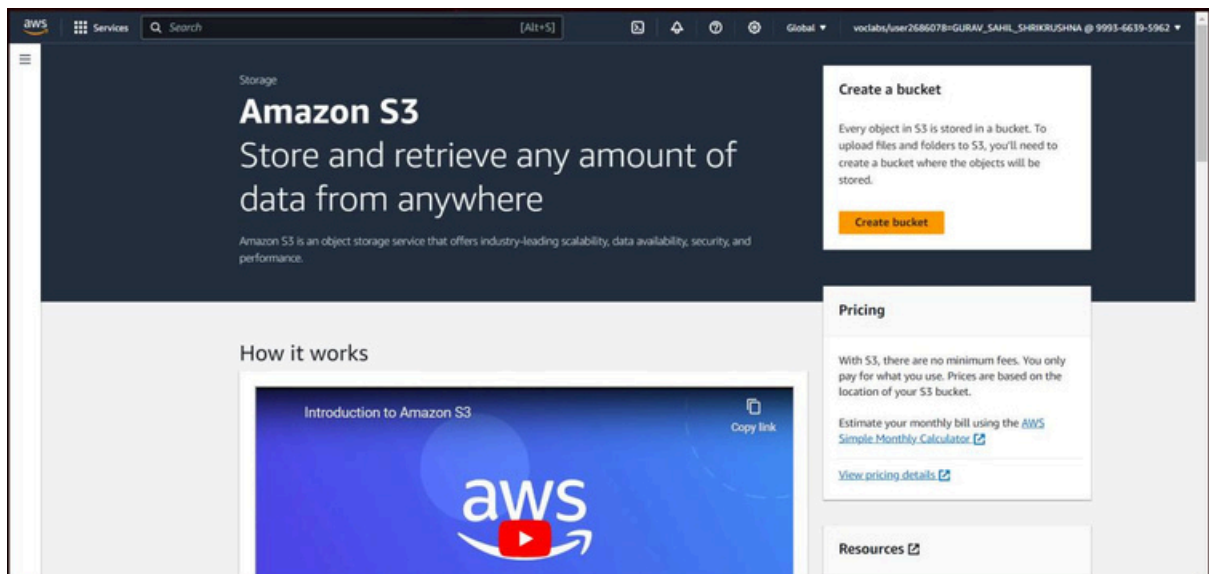


Fig 3

Name the bucket “sahil233” and select the AWS region “US East (N.Virginia)us-east-1” as shown in Fig 4.

Amazon S3 > Buckets > Create bucket

## Create bucket [Info](#)

Buckets are containers for data stored in S3. [Learn more](#)

### General configuration

Bucket name

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

AWS Region

Copy settings from existing bucket - *optional*  
Only the bucket settings in the following configuration are copied.

**Fig 4**

Object ownership determines who can specify access to objects. Select "ACL Enable" in Object Ownership. Select "Bucket owner preferred" in that as show in Fig 5. When "Bucket Owner Preferred" is enabled, the AWS account that uploaded the object becomes the owner of the object with full control over it, and can grant access to other users via ACL(Access Control Lists).

**Object Ownership** [Info](#)

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

**ACLs disabled (recommended)**  
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

**ACLs enabled**  
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

**⚠** We recommend disabling ACLs, unless you need to control access for each object individually or to have the object writer own the data they upload. Using a bucket policy instead of ACLs to share data with users outside of your account simplifies permissions management and auditing.

Object Ownership

**Bucket owner preferred**  
If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.


**Object writer**  
The object writer remains the object owner.

**i** If you want to enforce object ownership for new objects only, your bucket policy must specify that the bucket-owner-full-control canned ACL is required for object uploads. [Learn more](#) [↗](#)

**Fig 5**

As shown in Fig 6, "Block public access settings for this bucket" is disabled.

## Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#) 

### Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

#### Block public access to buckets and objects granted through *new* access control lists (ACLs)

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

#### Block public access to buckets and objects granted through *any* access control lists (ACLs)

S3 will ignore all ACLs that grant public access to buckets and objects.

#### Block public access to buckets and objects granted through *new* public bucket or access point policies

S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

#### Block public and cross-account access to buckets and objects through *any* public bucket or access point policies

S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.



#### Turning off block all public access might result in this bucket and the objects within becoming public

AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Fig 6

Keep "Bucket Versioning" disable as shown in Fig 7.

### Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning

Disable

Enable

---

### Tags - optional (0)

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

No tags associated with this bucket.

[Add tag](#)

Fig 7

Keep the default encryption as shown in Fig 8 and enable "Bucket Keys".

### Default encryption [Info](#)

Server-side encryption is automatically applied to new objects stored in this bucket.

---

Encryption type [Info](#)

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Server-side encryption with AWS Key Management Service keys (SSE-KMS)

Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)  
Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing](#) on the [Storage](#) tab of the [Amazon S3 pricing page](#).

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

Disable

Enable

---

► **Advanced settings**

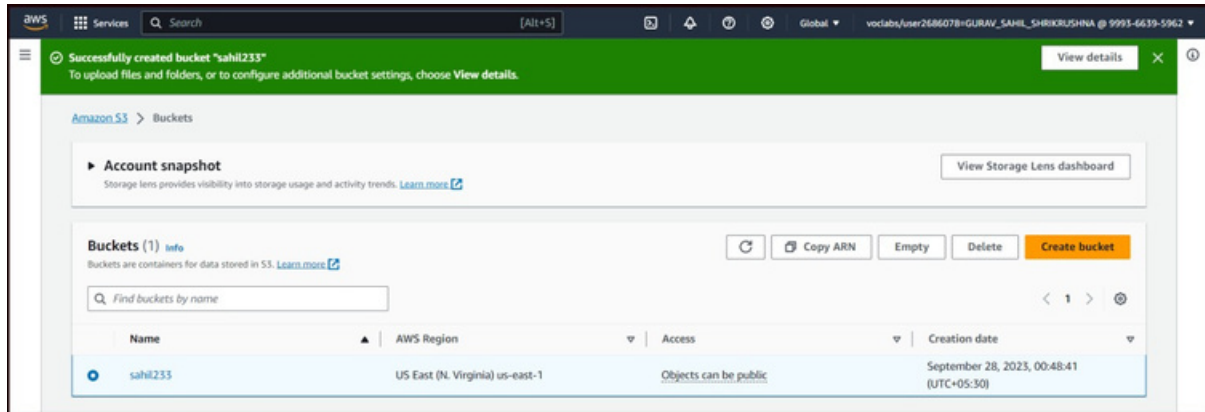
---

[i](#) After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel [Create bucket](#)

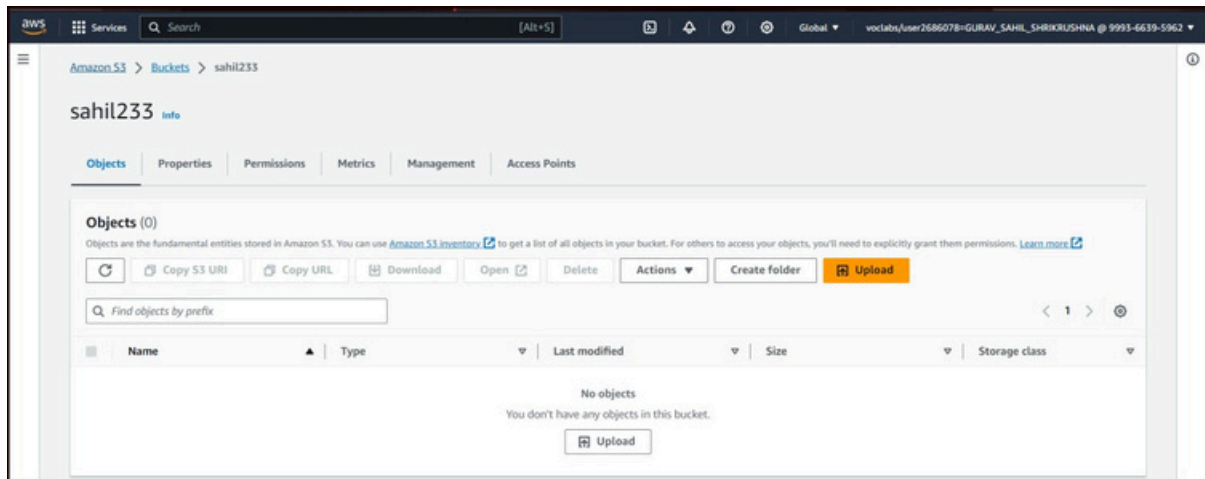
**Fig 8**

In Fig 9 we can see that the bucket has been created successfully. Then select Bucket and click on that bucket.



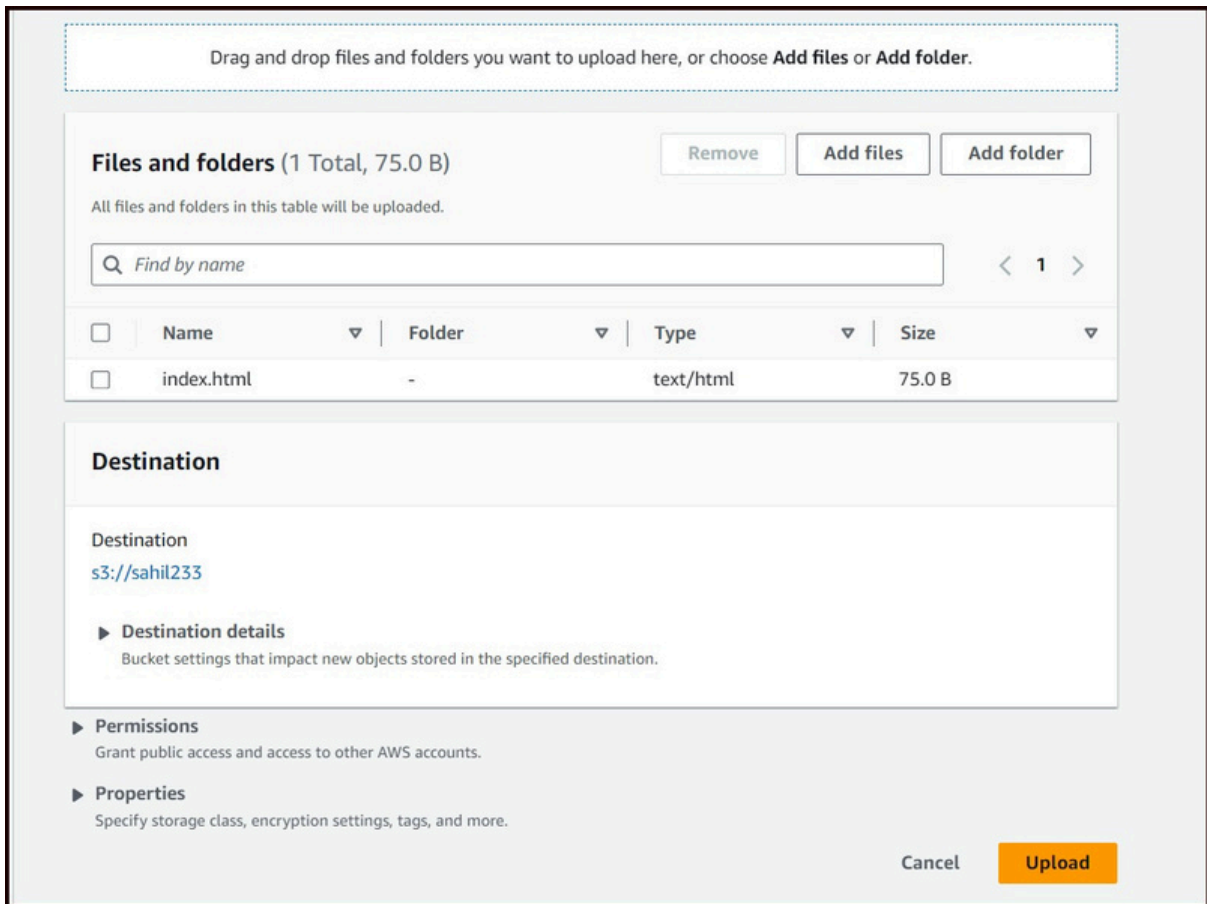
**Fig 9**

After clicking on Create Bucket we get the interface as shown in Fig 10. Then click on the "Upload" button and upload html file.



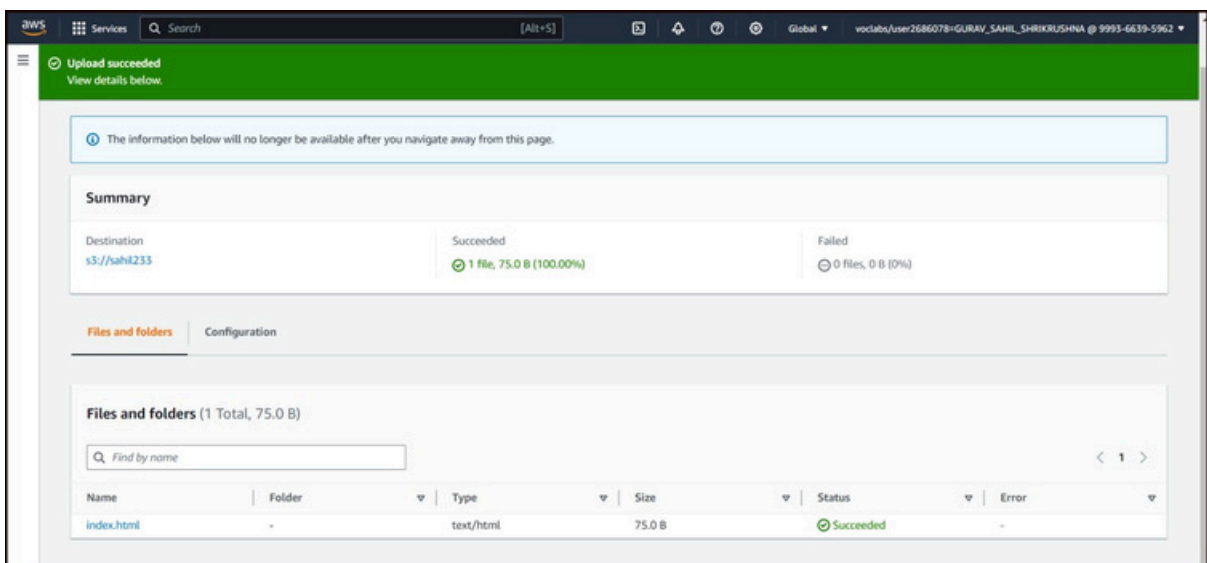
**Fig 10**

In Fig 11 we see that the index.html file has been uploaded. In other keep it as it is and click on "Upload" button.



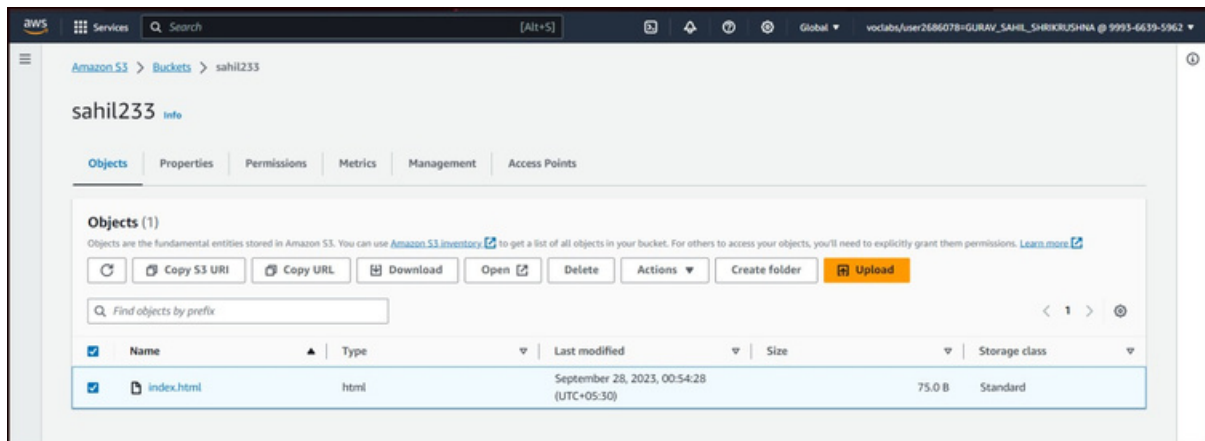
**Fig 11**

In Fig 12 we see that the index.html file has been uploaded successfully.



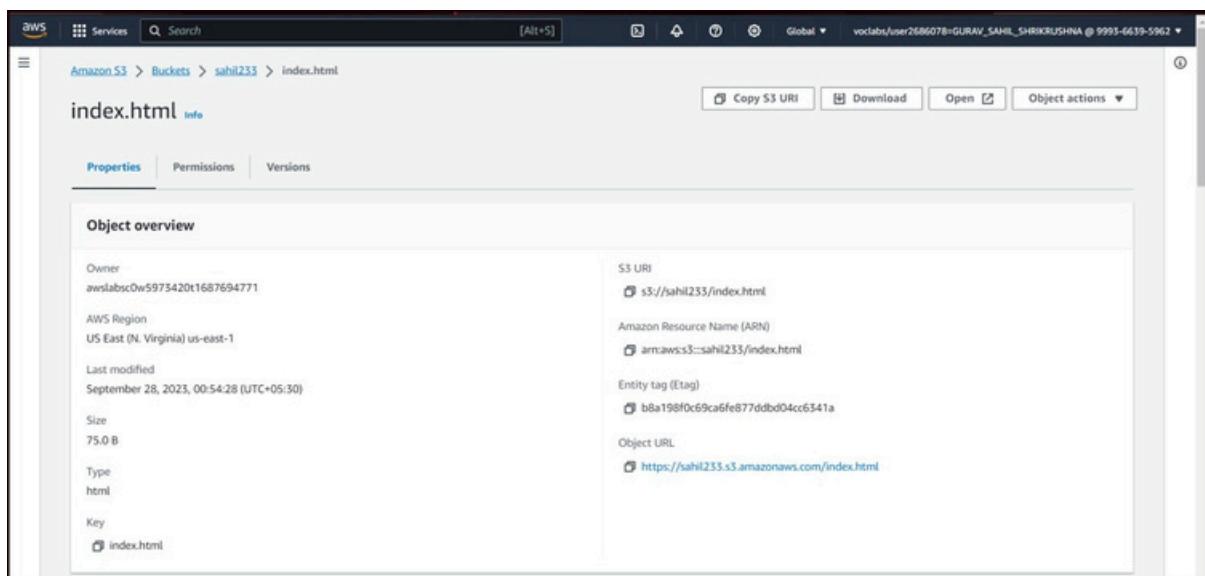
**Fig 12**

Then open the created bucket and select the uploaded file in it as shown in Fig 13. Then click on selected image.



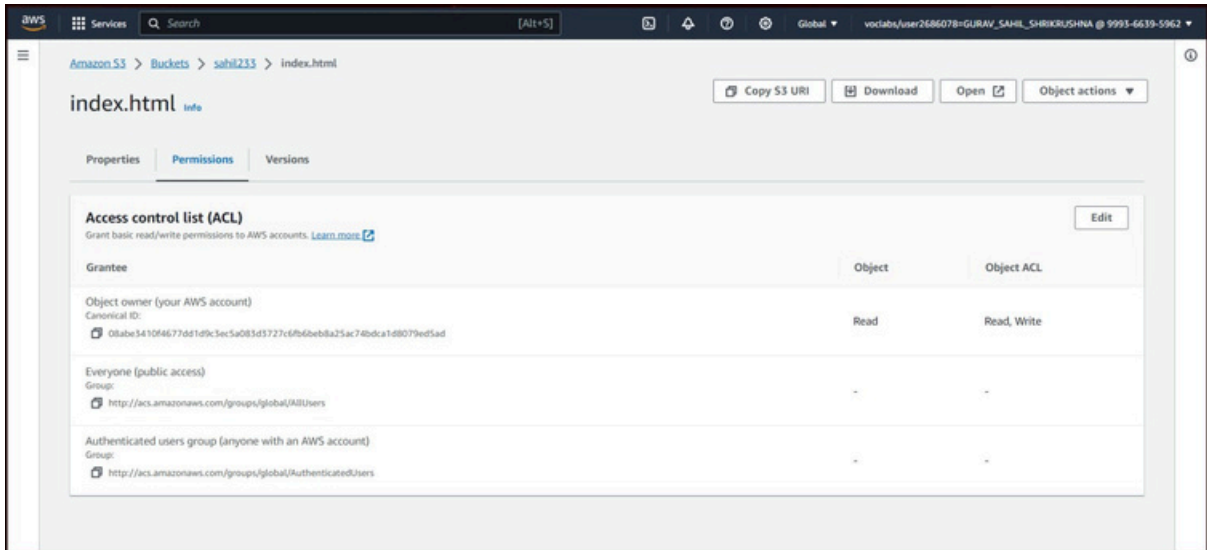
**Fig 13**

After clicking on the image we get the interface as shown in Fig 14 then click on "Permission" tab.



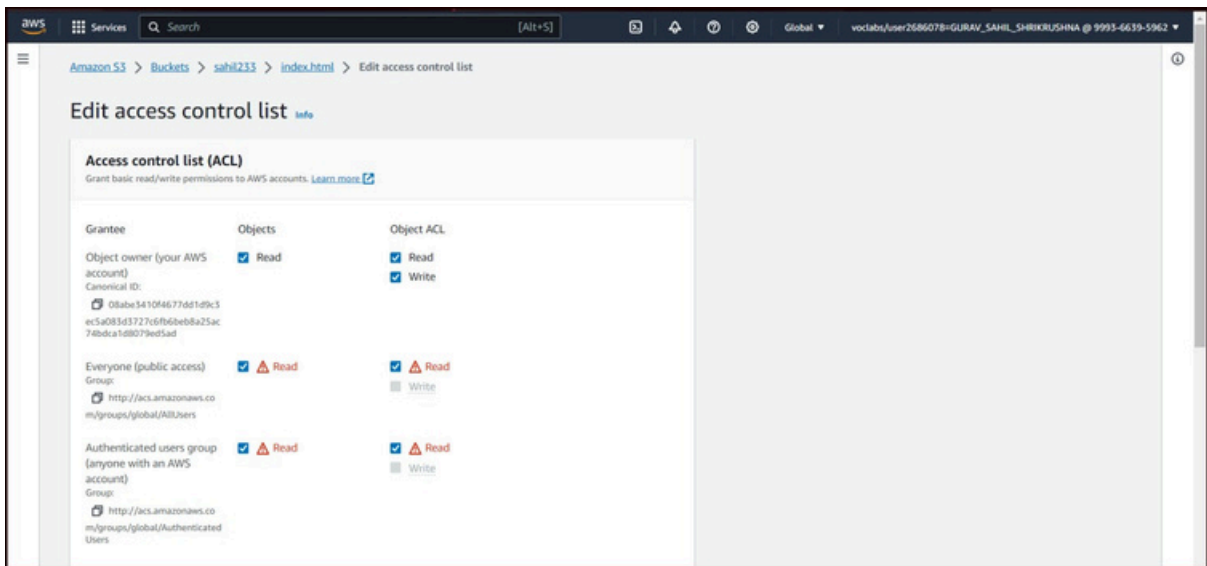
**Fig 14**

In Fig 15, we can see the details of the Permissions tab. Then click "Edit" in the Access Control List (ACL).



**Fig 15**

Enable public access and authenticate the user group in "Edit access control list" as shown in Fig 16



**Fig 16**

Keep the others as they are in "Edit Access Control List" and click "Save Changes" as shown in Fig 17.

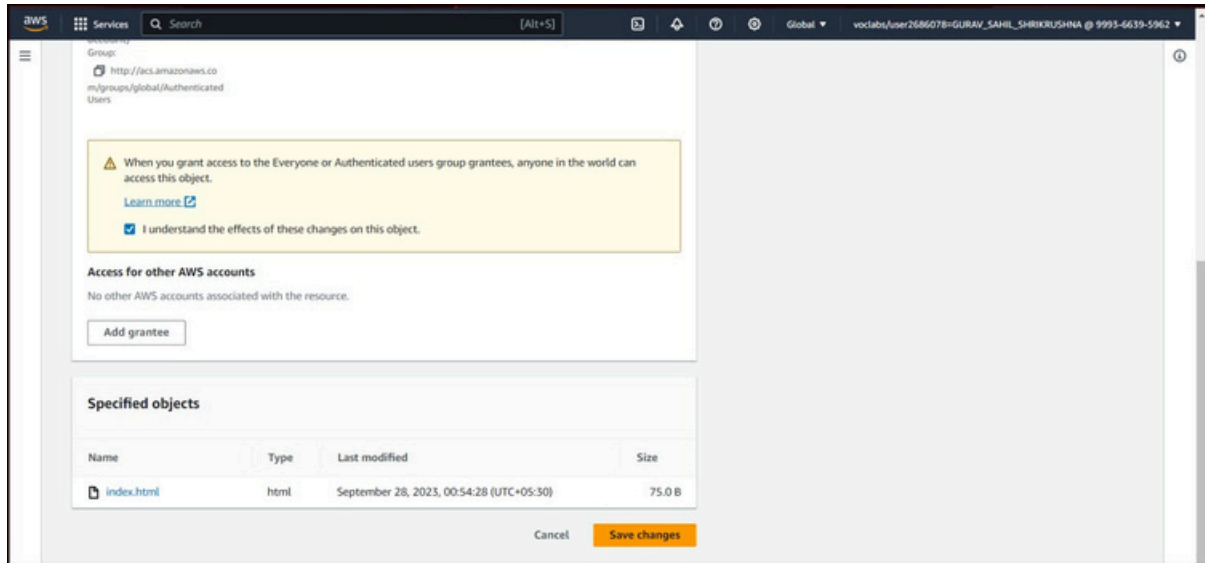


Fig 17

Then reopen the created bucket and open the uploaded file. Copy the "Object URL" as shown in Fig 18.

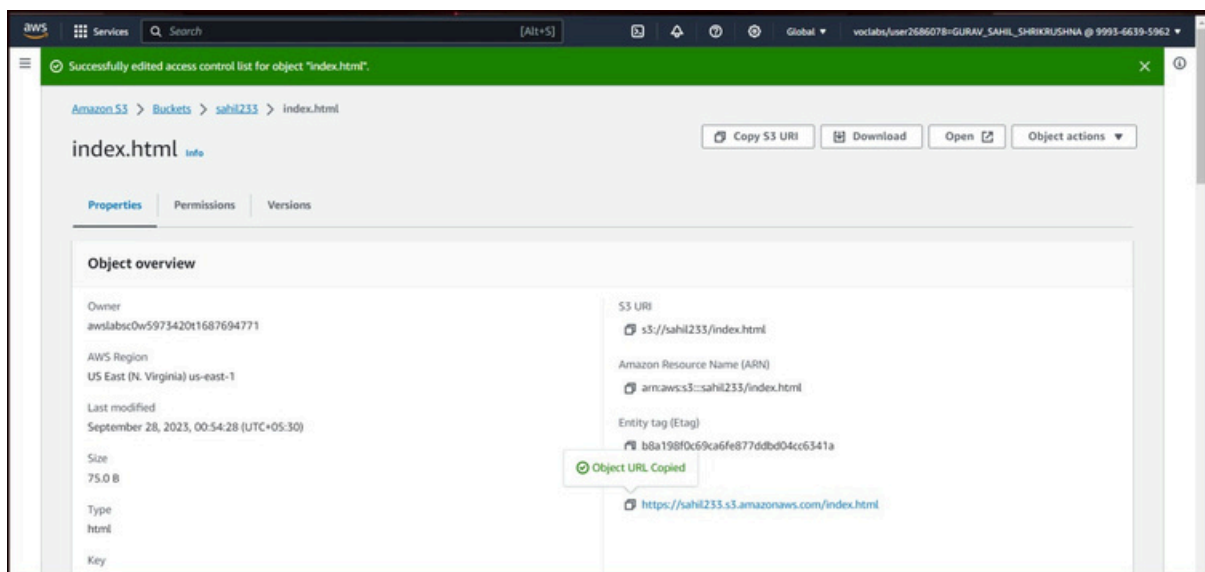
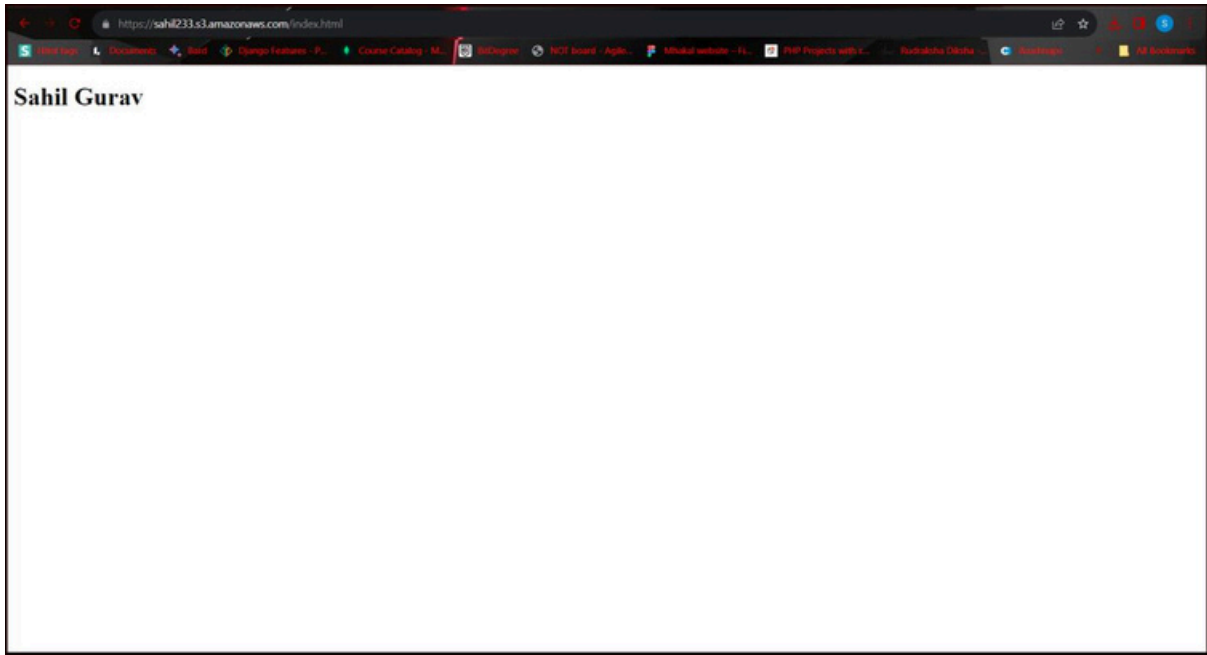


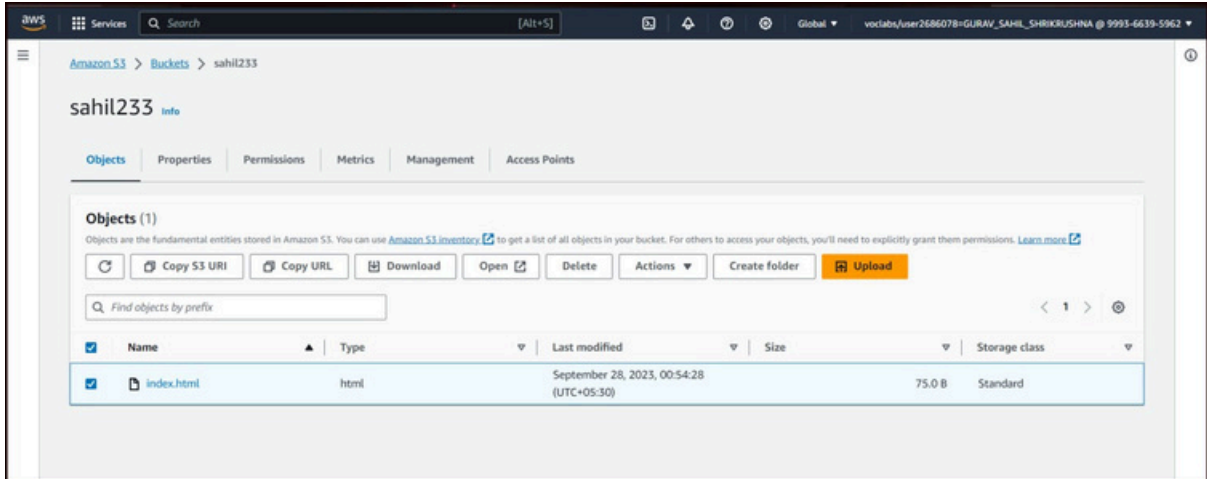
Fig 18

Paste the copied "Object URL" on the new tab as shown in Fig 19 and we can see the output of the static web page.



**Fig 19**

To delete a bucket, first delete the files inside it, for this select the uploaded file and click on "Delete" button as shown in Fig 20 .



**Fig 20**

In Fig 21 we can see that the object has been successfully deleted.

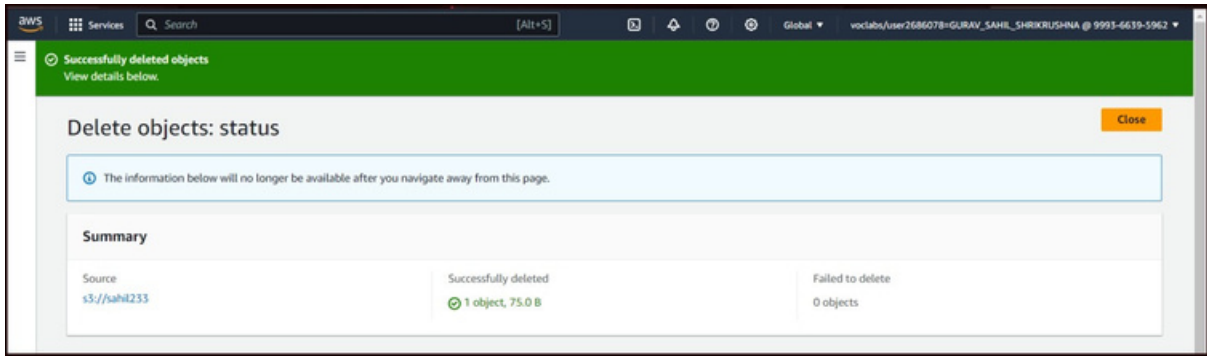


Fig 21

Then select the created bucket and click on “Delete” button to delete the bucket as shown in Fig 22

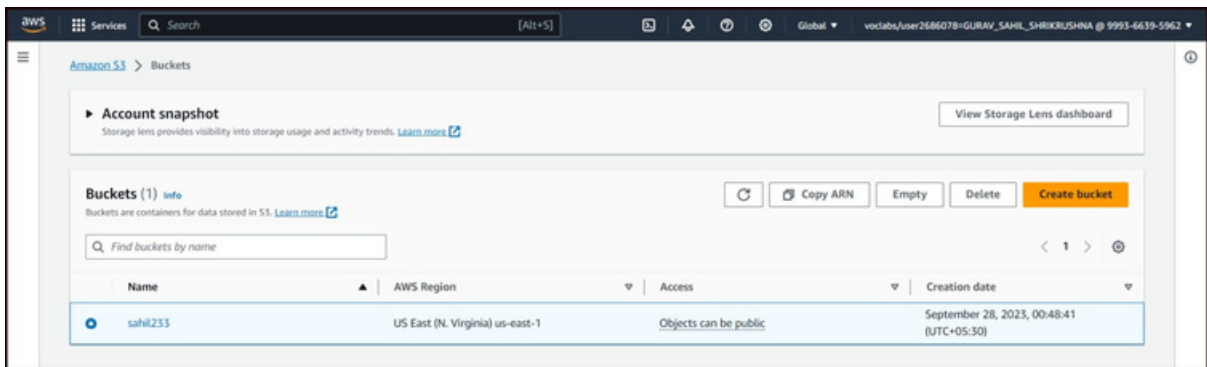
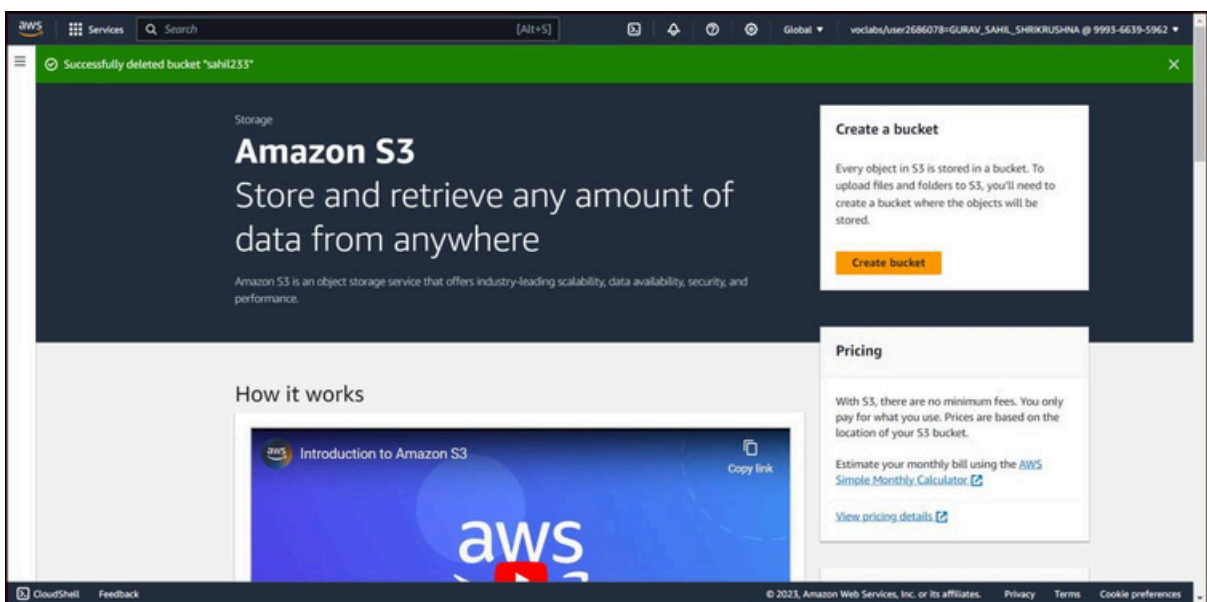


Fig 22

In Fig 23 we can see that the created bucket has been deleted successfully



**Fig 23**