

## Experiment No. 1A

### Static Hosting

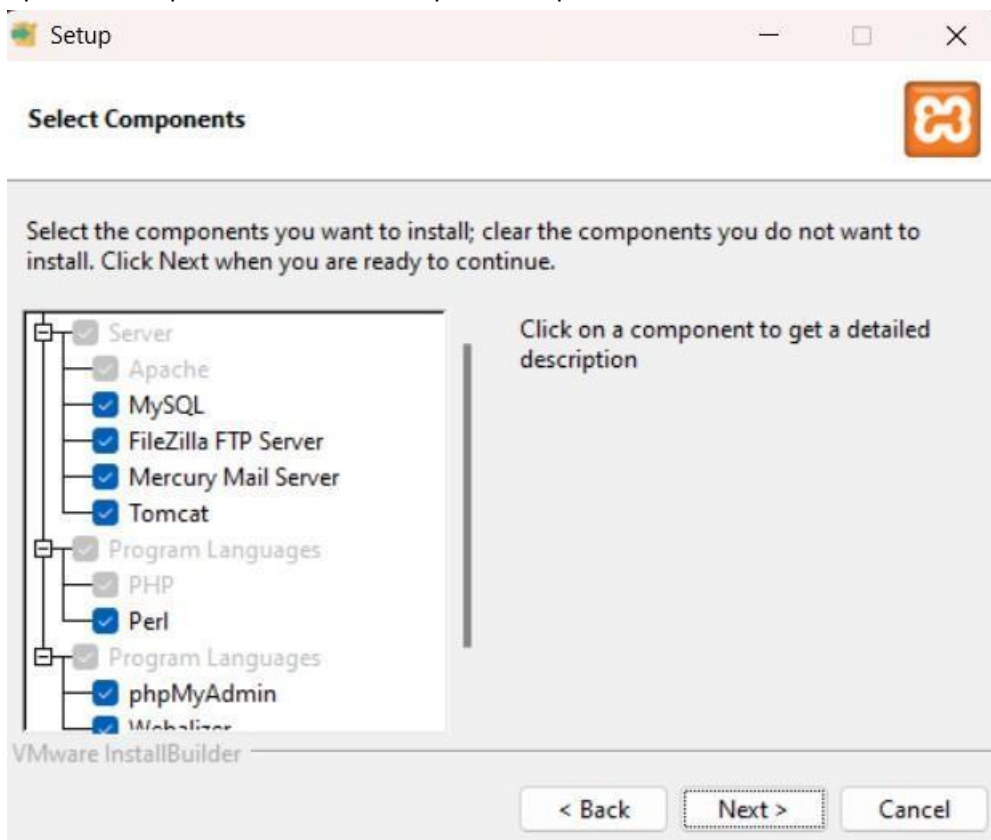
#### On local server (XAMPP)

Step 1: Install XAMPP from <https://www.apachefriends.org/>

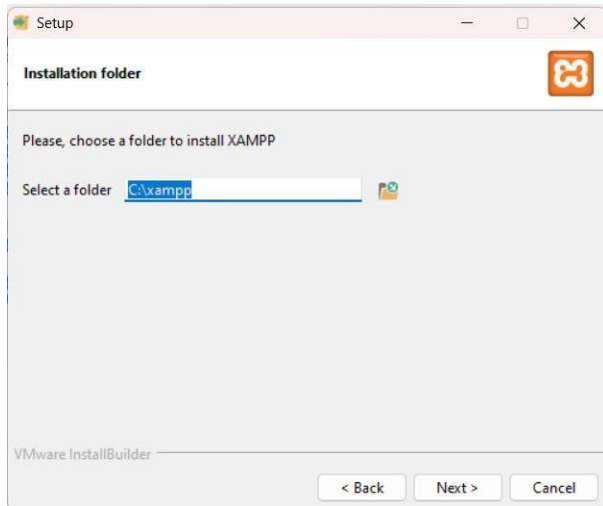
- 1) Select your OS. It will automatically start downloading.



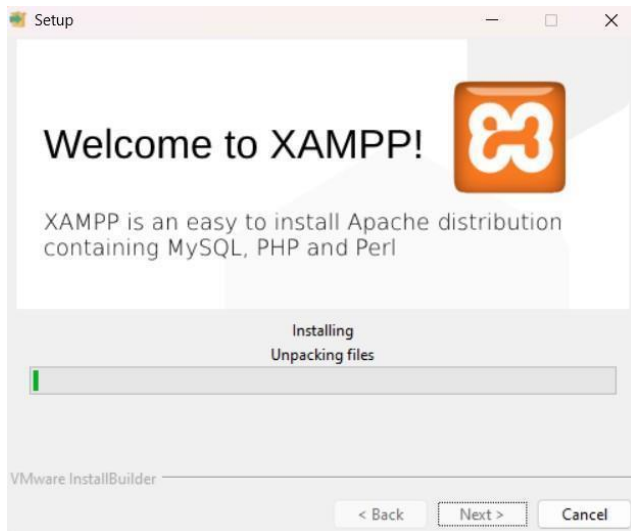
- 2) Open the setup file. Select all the required components and click next



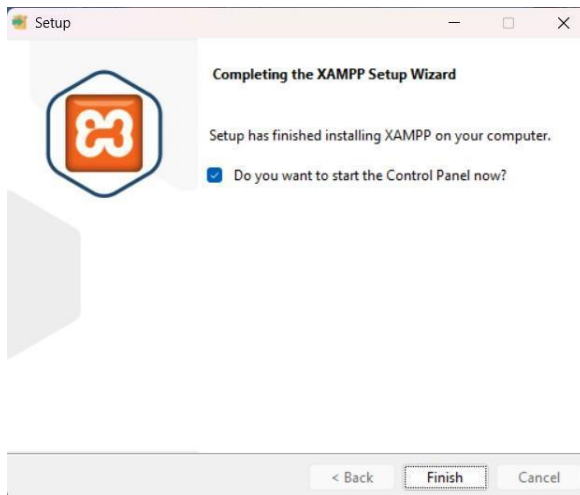
- 3) Choose the folder to install XAMPP in. Make sure the folder is empty. Click next



- 4) Select the language, click next. XAMPP starts to install



- 5) The installation is complete. Click Finish



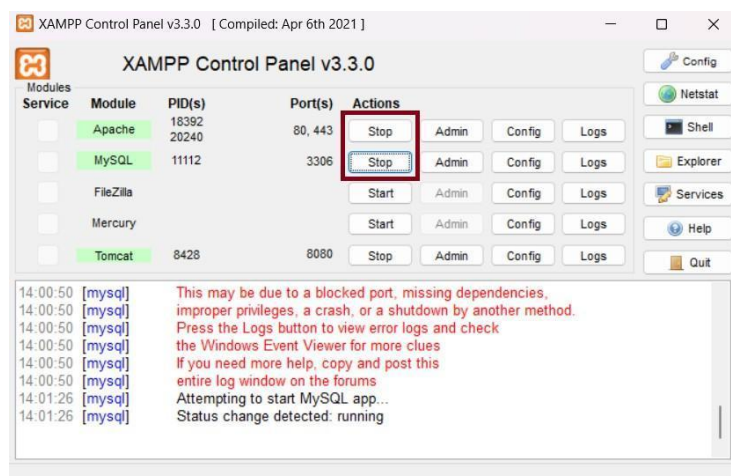
**Step 2:** Setup a file that is to be hosted on the server. Make sure the file has extension **.php**



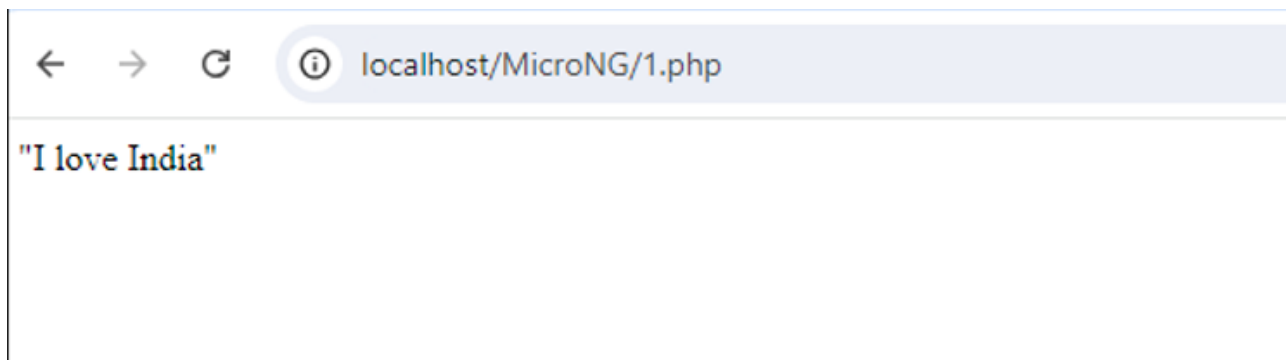
**Step 3:** Go to the directory where XAMPP was installed. Go to **htdocs** folder. Place your folder in this directory.

Name	Date modified	Type	Size
dashboard	06-08-2024 20:42	File folder	
img	06-08-2024 20:42	File folder	
webalizer	06-08-2024 20:42	File folder	
xampp	06-08-2024 22:44	File folder	
applications	15-06-2022 21:37	Chrome HTML Do...	4 KB
bitnami	15-06-2022 21:37	CSS Source File	1 KB
favicon.ico	16-07-2015 21:02	ICO File	31 KB
index	16-07-2015 21:02	PHP Source File	1 KB
1	06-08-2024 22:48	PHP Source File	1 KB
text	06-08-2024 22:23	PHP Source File	1 KB

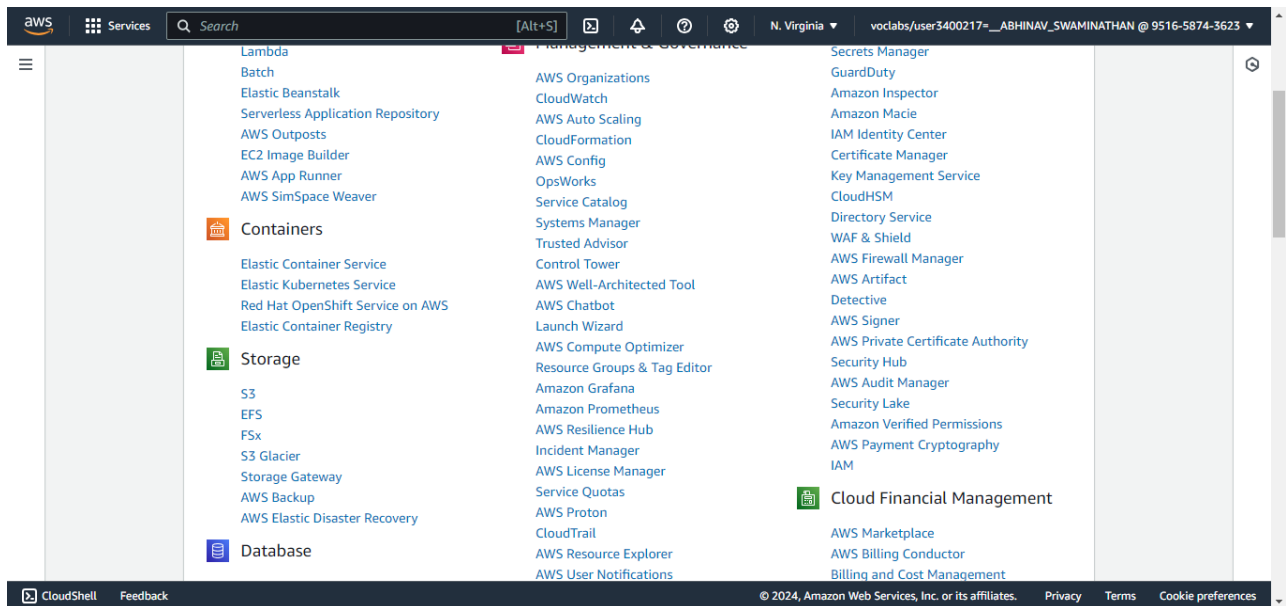
**Step 4:** Open XAMPP Control Panel, start the Apache service (Required) and mySQL service (if needed)



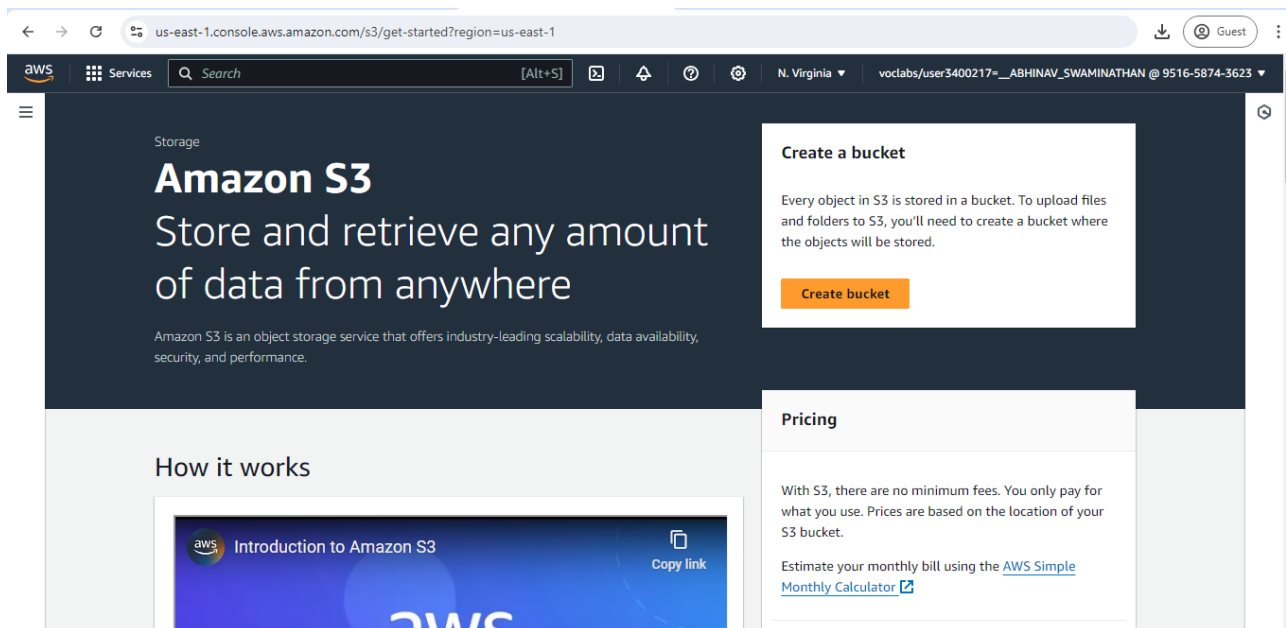
**Step 5:** Open your web browser. Type localhost/YOUR\_FILENAME.php. This will open your website on your browser.



# AWS S3



**Step 1:** Login to your AWS account. Go to services and open **S3**.



## Step 2: Click on Create Bucket

Amazon S3 > Buckets > Create bucket

### Create bucket [Info](#)

Buckets are containers for data stored in S3.

#### General configuration

AWS Region  
US East (N. Virginia) us-east-1

Bucket type [Info](#)

☒ **General purpose**  
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory - New**  
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

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

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

## Step 3: Give a name to your bucket, keeping other options default, scroll down and click on Create Bucket

AWS Region  
US East (N. Virginia) us-east-1

Bucket type [Info](#)

☒ **General purpose**  
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory - New**  
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

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

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

Format: s3://bucket/prefix

#### 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.

#### Step 4: Click on the name of your bucket and goto Properties

The screenshot displays the AWS Management Console interface for the 'abhinav215' S3 bucket. The top navigation bar includes the AWS logo, 'Services' menu, a search bar, and user information. The breadcrumb trail shows 'Amazon S3 > Buckets > abhinav215'. The bucket name 'abhinav215' is prominently displayed with an 'Info' link. Below this, a tabbed interface shows 'Objects', 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. The 'Properties' tab is selected, showing a 'Bucket overview' section with details: AWS Region (US East (N. Virginia) us-east-1), Amazon Resource Name (ARN) (arn:aws:s3:::abhinav215), and Creation date (August 7, 2024, 09:22:15 (UTC+05:30)). The 'Bucket Versioning' section shows it is 'Disabled' and includes a 'Multi-factor authentication (MFA) delete' option. An 'Edit' button is visible next to the versioning status. The bottom of the console shows a 'CloudShell' button, a 'Feedback' link, and a footer with copyright information and links to Privacy, Terms, and Cookie preferences.

aws Services Search [Alt+S] N. Virginia voclabs/user3400217-\_\_ABHINAV\_SWAMINATHAN @ 9516-5874-3623

Amazon S3 > Buckets > abhinav215

abhinav215 Info

Objects Properties Permissions Metrics Management Access Points

**Objects (0) Info**

Copy S3 URI Copy URL Download Open Delete Actions Create folder Upload

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

No objects  
You don't have any objects in this bucket.

Upload

Amazon S3 > Buckets > abhinav215

abhinav215 Info

Objects Properties Permissions Metrics Management Access Points

**Bucket overview**

AWS Region US East (N. Virginia) us-east-1	Amazon Resource Name (ARN) arn:aws:s3:::abhinav215	Creation date August 7, 2024, 09:22:15 (UTC+05:30)
---	---	---

**Bucket Versioning** Edit

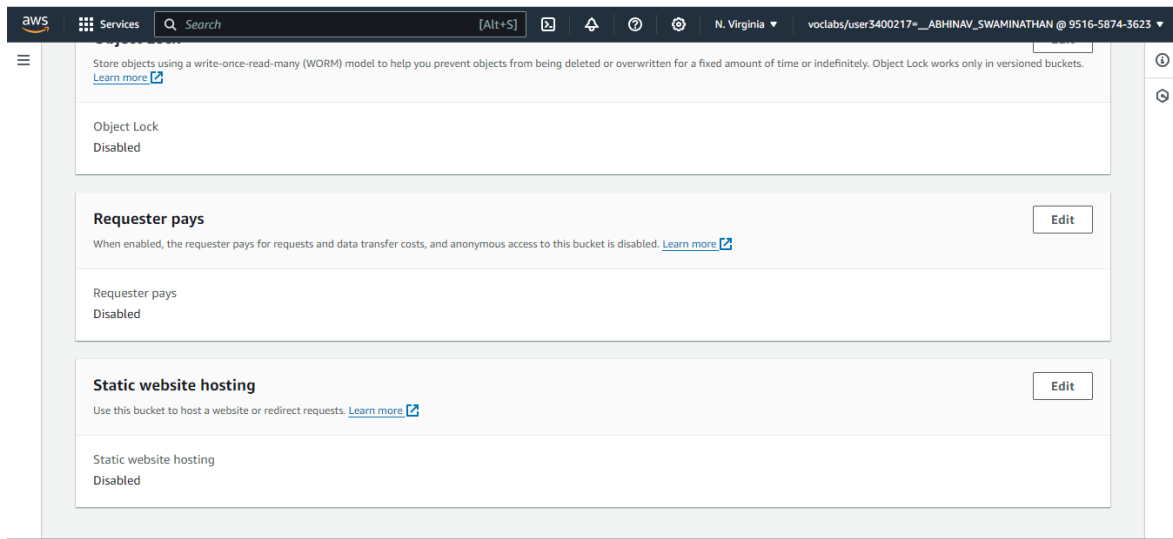
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures.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

Bucket Versioning  
Disabled  
Multi-factor authentication (MFA) delete

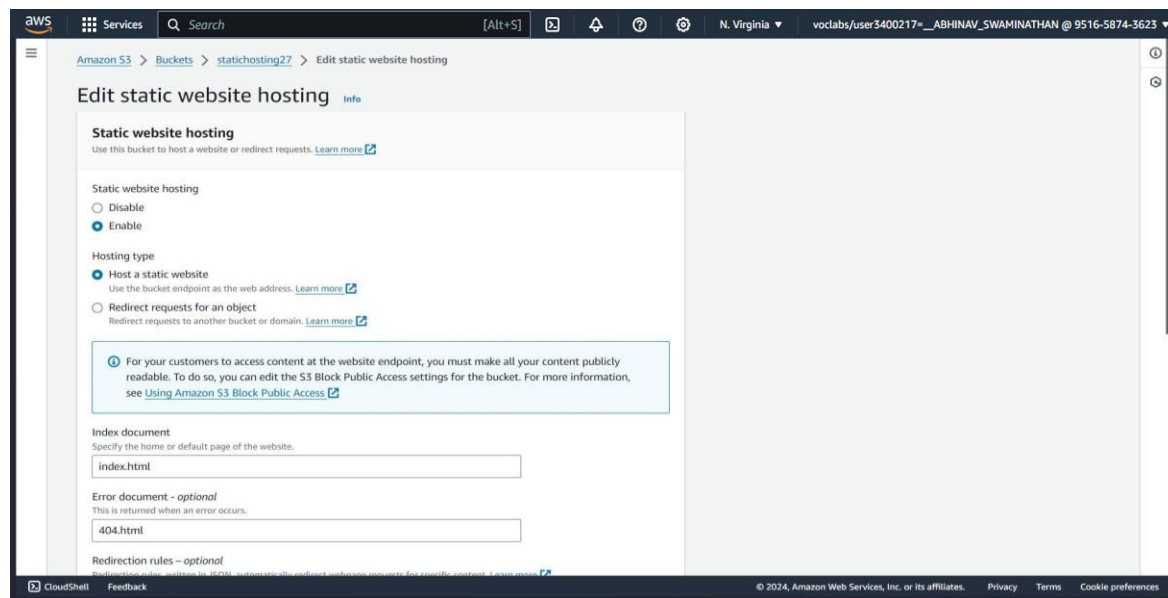
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

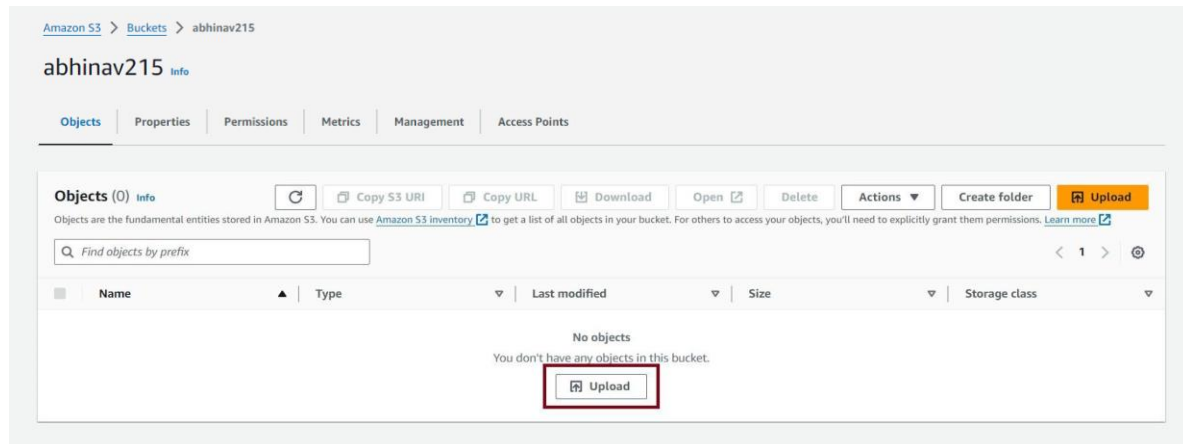
**Step 5:** Scroll down till you find Static website hosting, click on edit



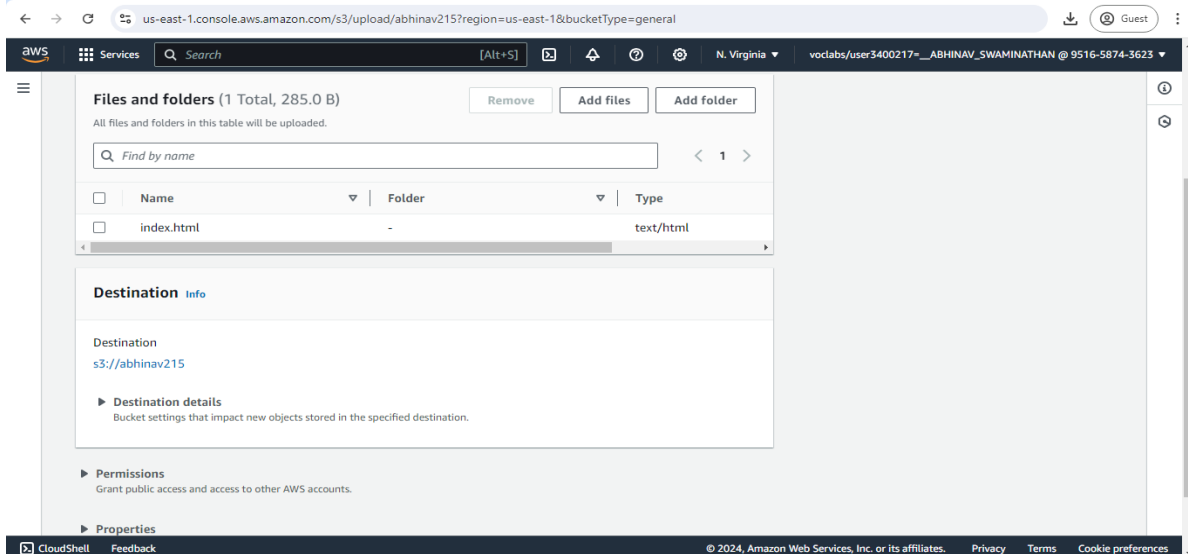
**Step 6:** Enable static website hosting, in Index document, write the name of your document and in error document, give name as 404.html. Save your changes.



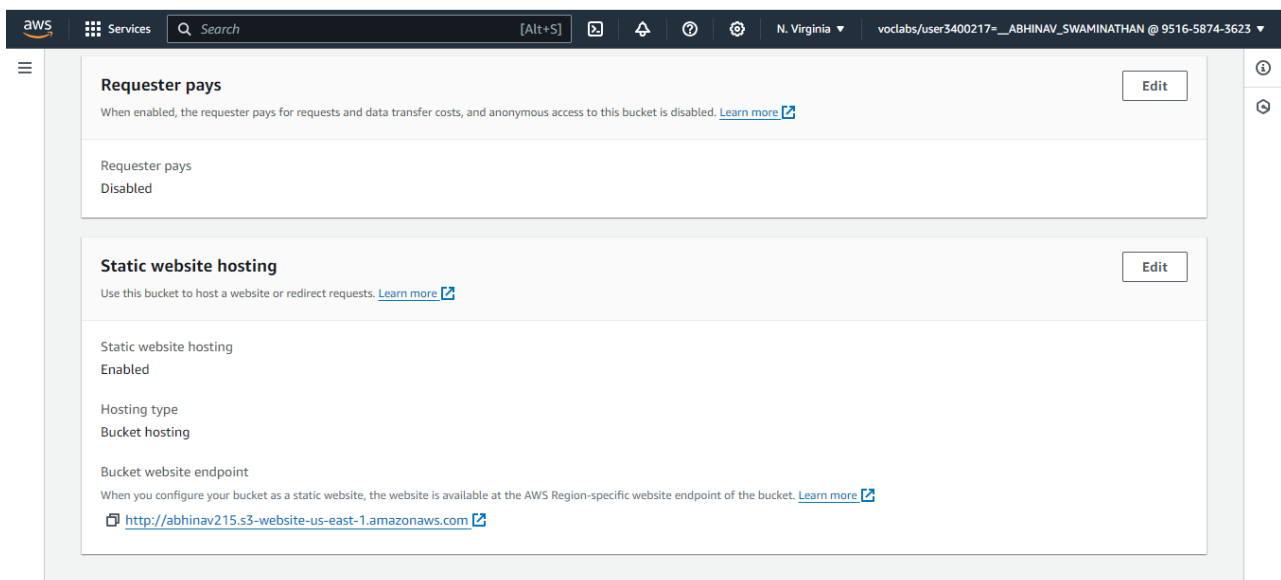
**Step 7:** Go to Objects tab and click on upload file.



**Step 8:** Click on Add files. Add all the files you want to upload. Then scroll down and click on Upload

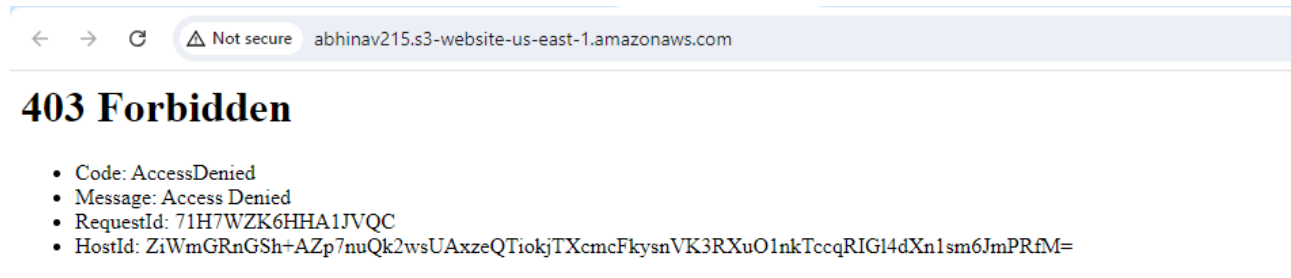


**Step 9:** This will take you to the Objects screen. Switch to Properties, scroll down to Static web hosting. There you would find the link (Bucket website endpoint) to your website.

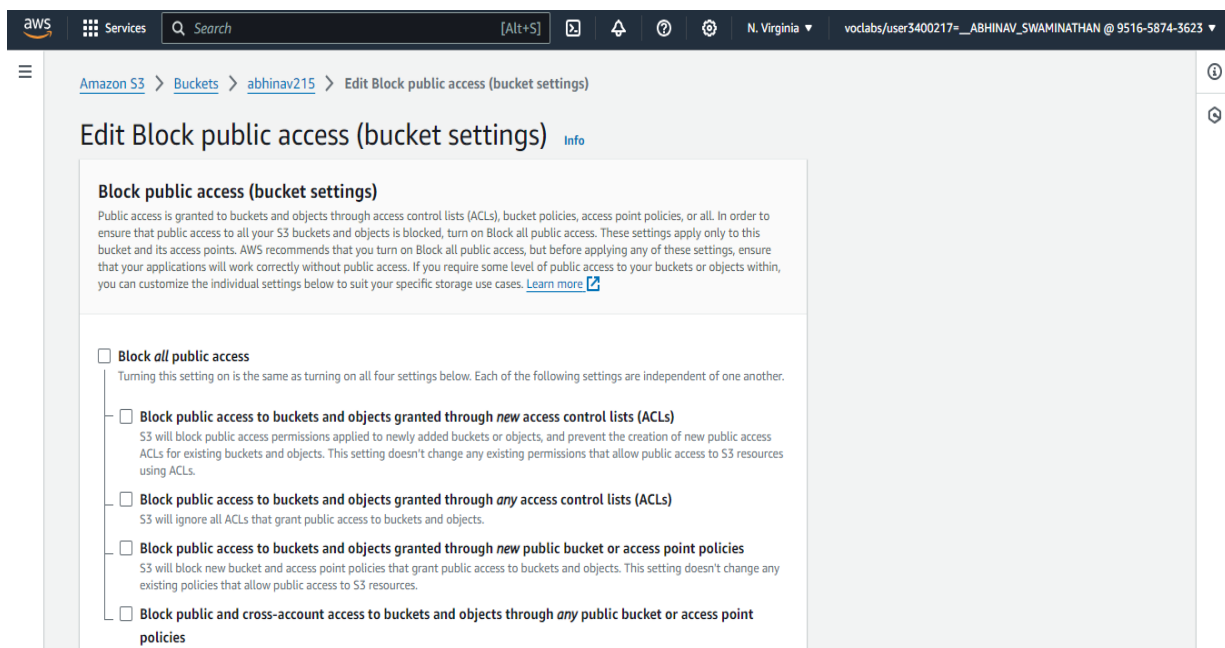




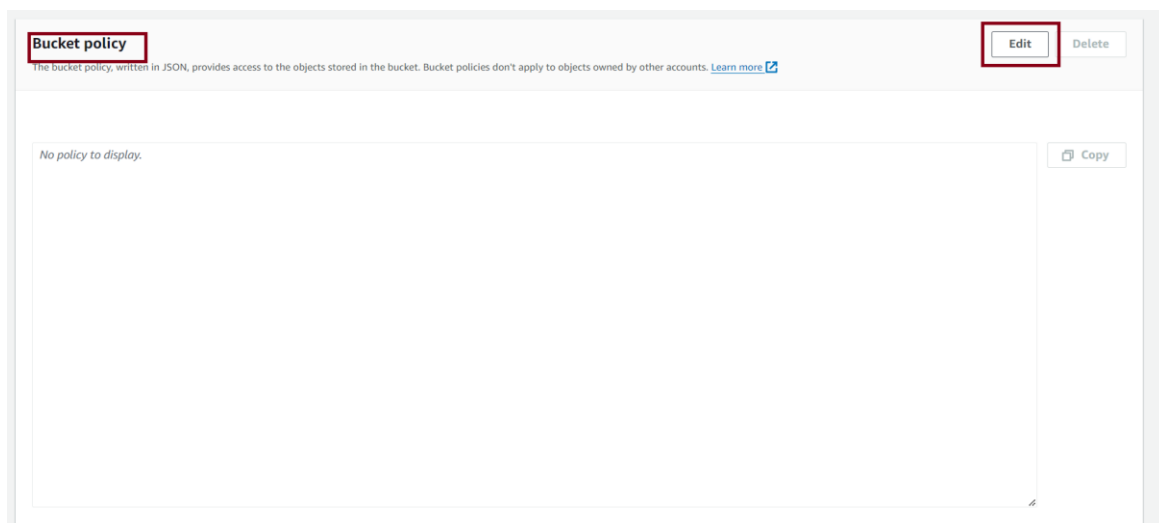
**Step 10:** Open the link. It will show a 403 forbidden error screen as the contents of the bucket are not available for the public users. To change this, go to Permissions tab, go to Block public access and click on edit



**Step 11:** Uncheck the Block all public access checkbox and click on save changes



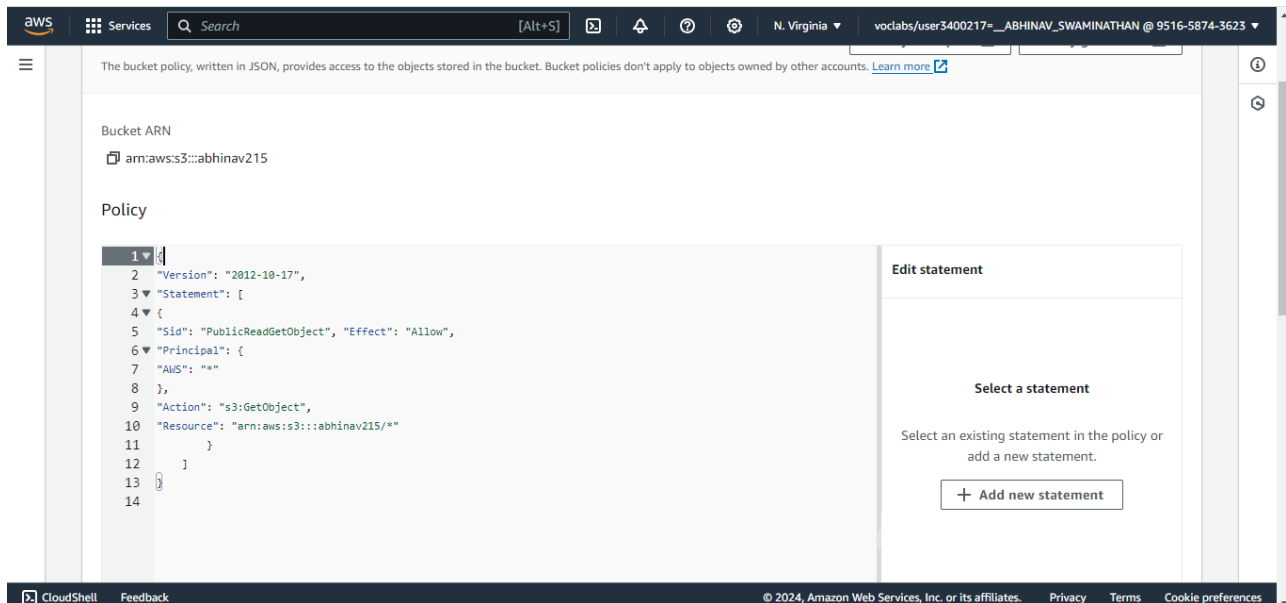
**Step 12:** Scroll down to bucket policy and click edit



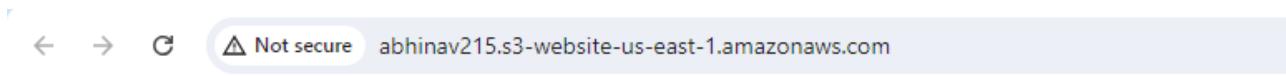
### Step 13:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadGetObject",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::YOUR-BUCKET-NAME-HERE/*"
    }
  ]
}
```

Paste this code snippet in the policy textarea. Replace YOUR-BUCKET-NAME-HERE with the name you have given to your bucket. Save the changes.



### Step 14: Now reload the website. You can see your website



# Hello, World!

Welcome to my first HTML page.