

STATIC WEBSITE HOST USING S3 BUCKET

- First, we should create a bucket in S3.

Amazon S3 > Buckets > Create bucket

Create bucket [Info](#)

Buckets are containers for data stored in S3.

General configuration

AWS Region

Asia Pacific (Mumbai) ap-south-1

Bucket name [Info](#)

hostwebsite102

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.

Choose bucket

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.

☐ 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.

📘 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](#)

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.

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

EC2

S3

RDS

IoT

IAM

VPC

CloudWatch

Lambda

Simple Notification Service

DataSync

Systems Manager

RDS

Database Migration Service

Q Search

[Alt+S]

Mumbai

Amogh

No tags associated with this bucket.

Add tag

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)

Before you encrypt 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

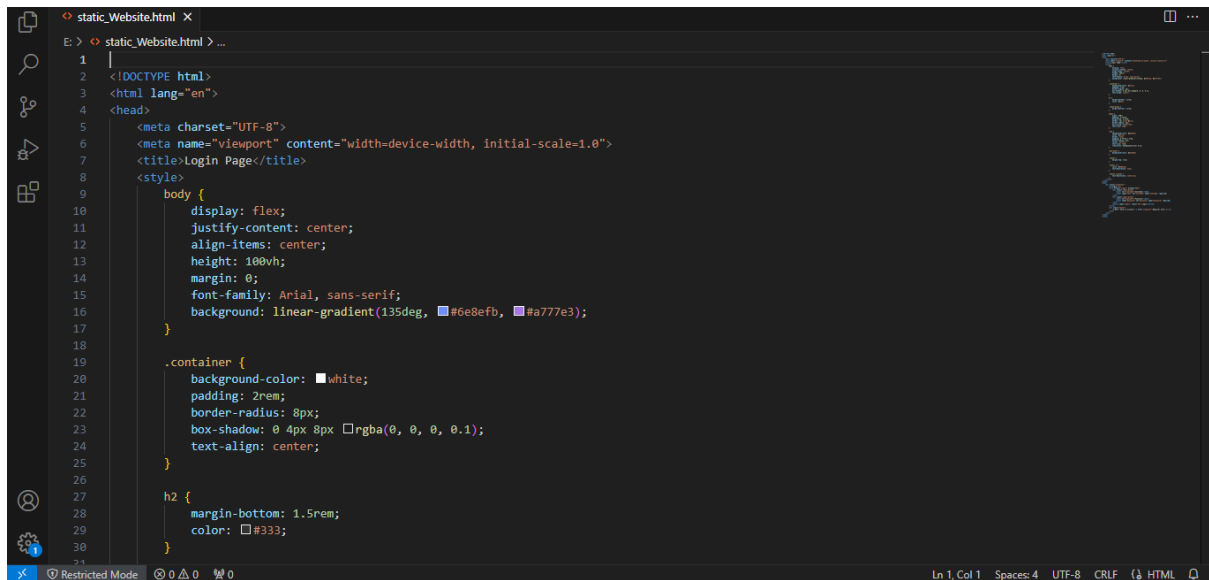
Advanced settings

📘 After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel

Create bucket

- After that we should write a code for the website in any code editor like VsCode, Sublime text, etc, or a notepad, and then upload in a folder
- Then we have to click on “Upload” and then upload the given folder or a file



```

1  |
2  <!DOCTYPE html>
3  <html lang="en">
4  <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Login Page</title>
8      <style>
9          body {
10             display: flex;
11             justify-content: center;
12             align-items: center;
13             height: 100vh;
14             margin: 0;
15             font-family: Arial, sans-serif;
16             background: linear-gradient(135deg, #6e8efb, #a777e3);
17         }
18
19         .container {
20             background-color: white;
21             padding: 2rem;
22             border-radius: 8px;
23             box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
24             text-align: center;
25         }
26
27         h2 {
28             margin-bottom: 1.5rem;
29             color: #333;
30         }
31     </style>

```

Html Code:

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Login Page</title>
```

```
<style>
```

```
body {
```

```
    display: flex;
```

```
    justify-content: center;
```

```
    align-items: center;
```

```
    height: 100vh;
```

```
    margin: 0;
```

```
    font-family: Arial, sans-serif;
```

```
    background: linear-gradient(135deg, #6e8efb, #a777e3);
```

```
}
```

```
.container {  
  background-color: white;  
  padding: 2rem;  
  border-radius: 8px;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
  text-align: center;  
}
```

```
h2 {  
  margin-bottom: 1.5rem;  
  color: #333;  
}
```

```
.input-group {  
  margin-bottom: 1.5rem;  
}
```

```
input {  
  width: 100%;  
  padding: 0.75rem;  
  margin-top: 0.5rem;  
  border: 1px solid #ccc;  
  border-radius: 4px;  
  box-sizing: border-box;  
  font-size: 1rem;  
}
```

```
.btn {  
  background-color: #6e8efb;  
  color: white;  
  border: none;  
  padding: 0.75rem 1.5rem;  
  border-radius: 4px;  
  cursor: pointer;
```

```
    font-size: 1rem;
    transition: background-color 0.3s;
}
```

```
.btn:hover {
    background-color: #5a76d5;
}
```

```
.footer {
    margin-top: 1rem;
}
```

```
.footer a {
    color: #6e8efb;
    text-decoration: none;
}
```

```
.footer a:hover {
    text-decoration: underline;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="container">
```

```
<h2>Login</h2>
```

```
<form action="/login" method="POST">
```

```
<div class="input-group">
```

```
<label for="username">Username</label>
```

```
<input type="text" id="username" name="username" required>
```

```
</div>
```

```
<div class="input-group">
```

```
<label for="password">Password</label>
```

```
<input type="password" id="password" name="password" required>
```

```
</div>
```

```
<button type="submit" class="btn">Login</button>
```

</form>

<div class="footer">

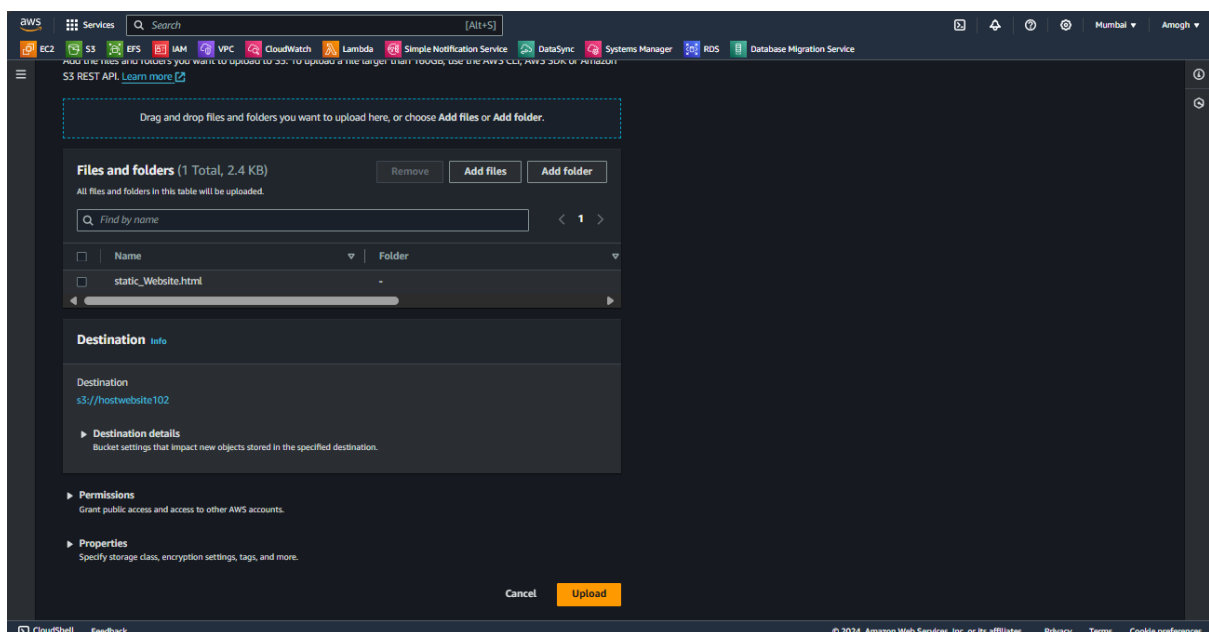
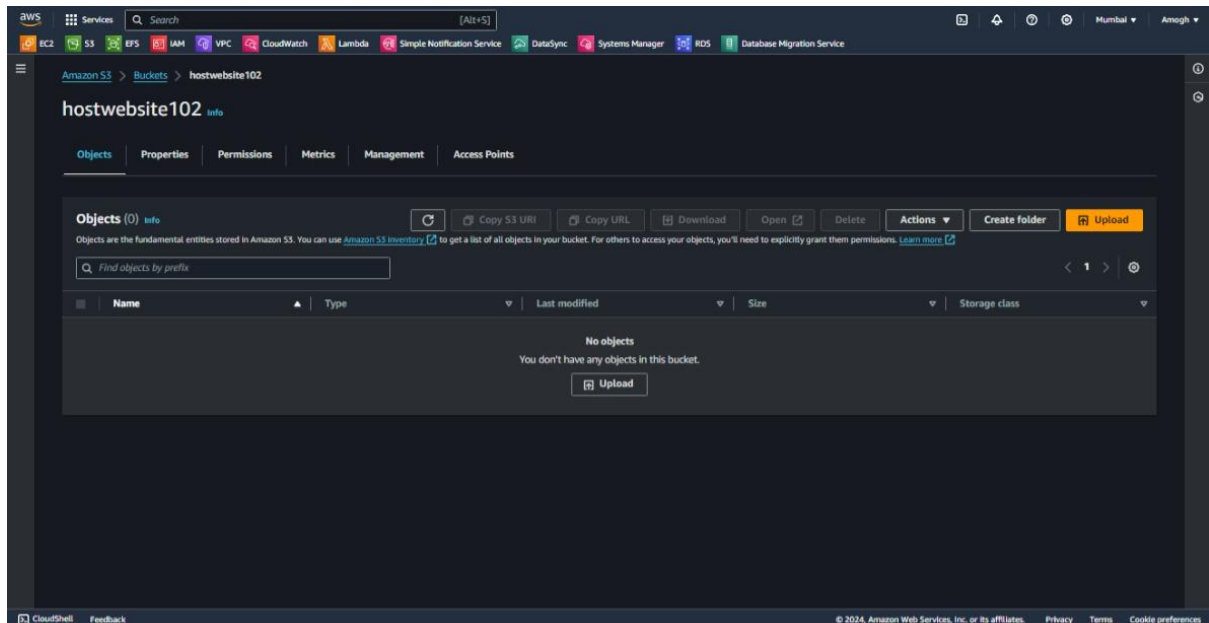
<p>Don't have an account? Register here</p>

</div>

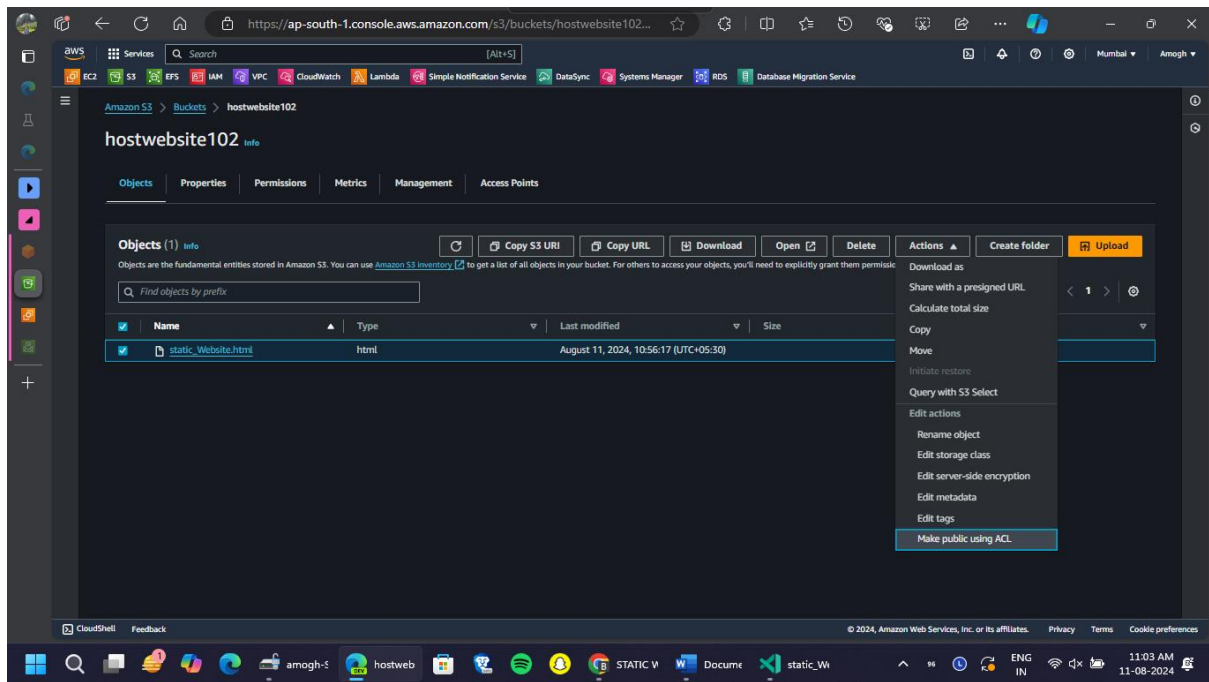
</div>

</body>

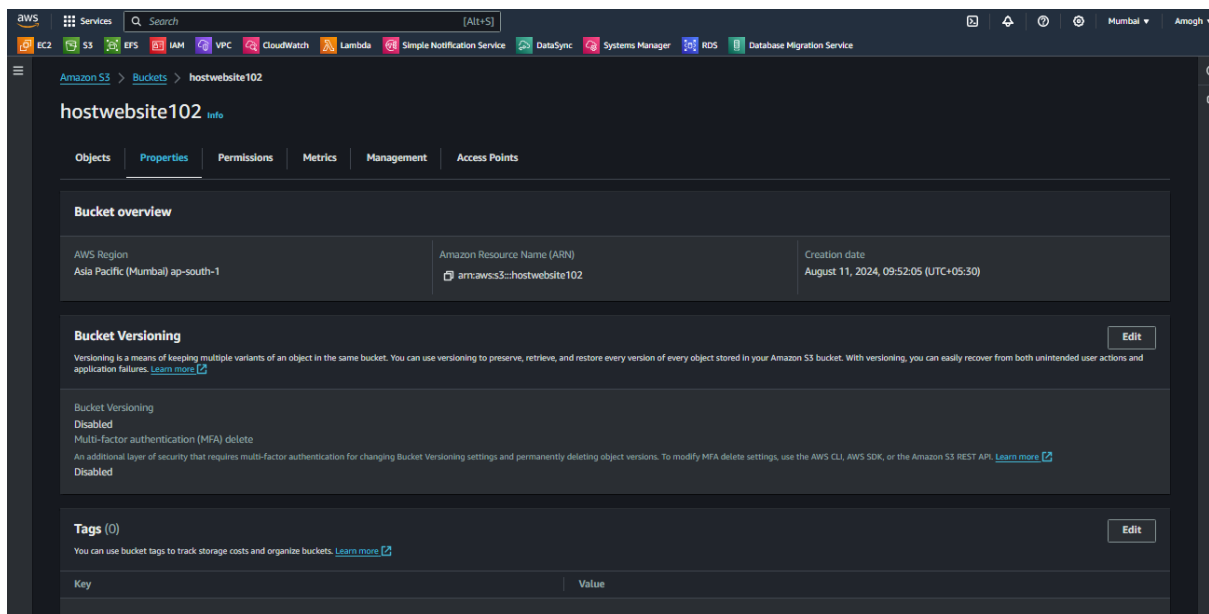
</html>



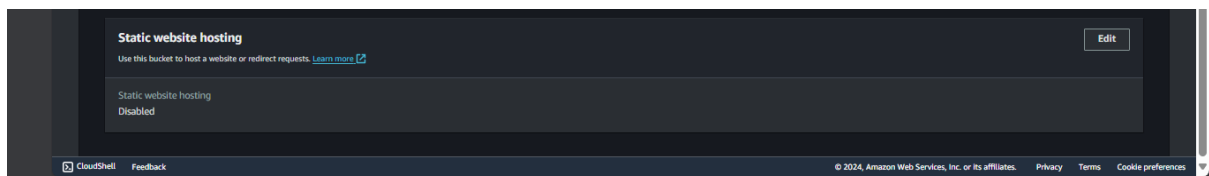
- Now we have successfully uploaded. So, we have to set the object policy by “**ACL**”
- We have to click on the file, then go to actions and select “**Making public using ACL**”.



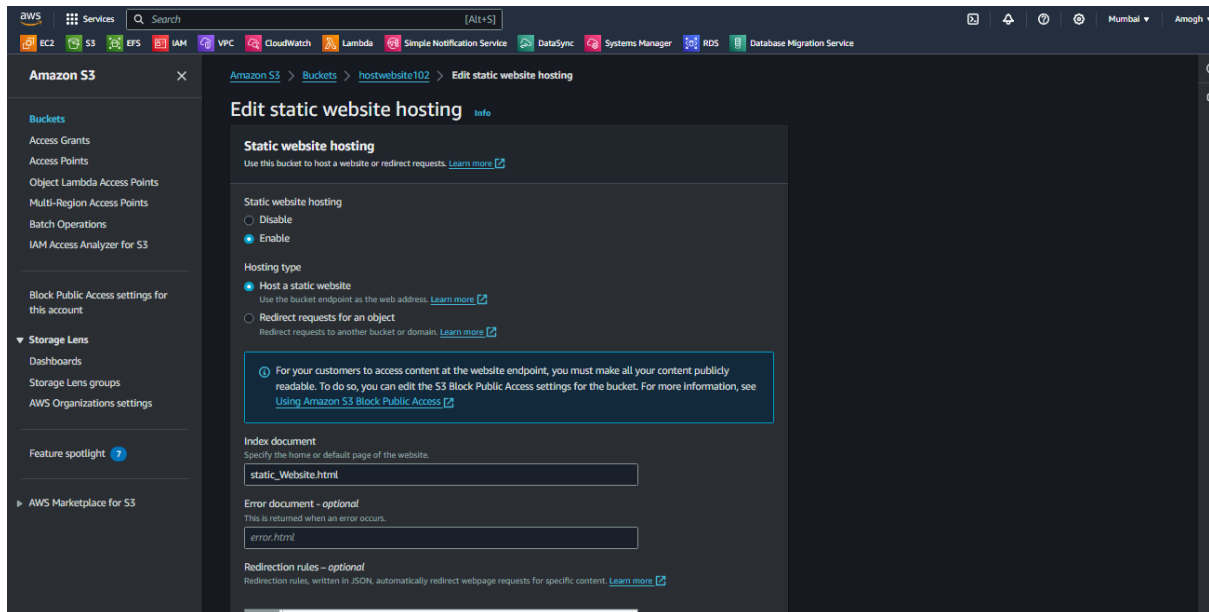
- Now go-to “Static Web Hosting” option in the Properties



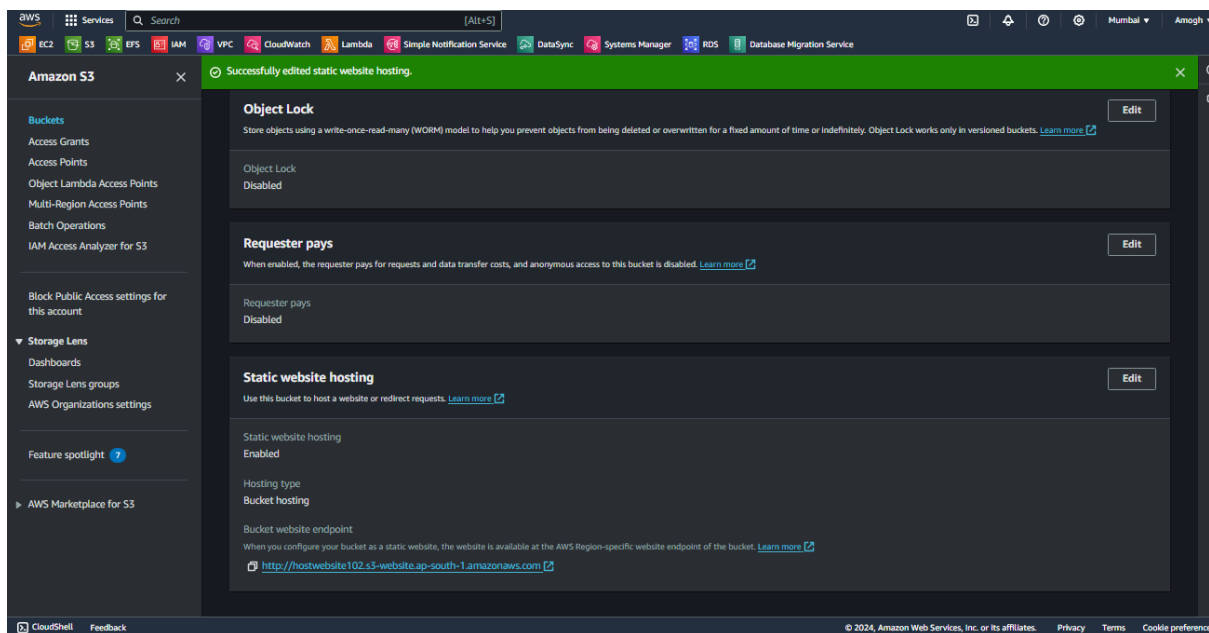
- Edit with enable the option “Enable”



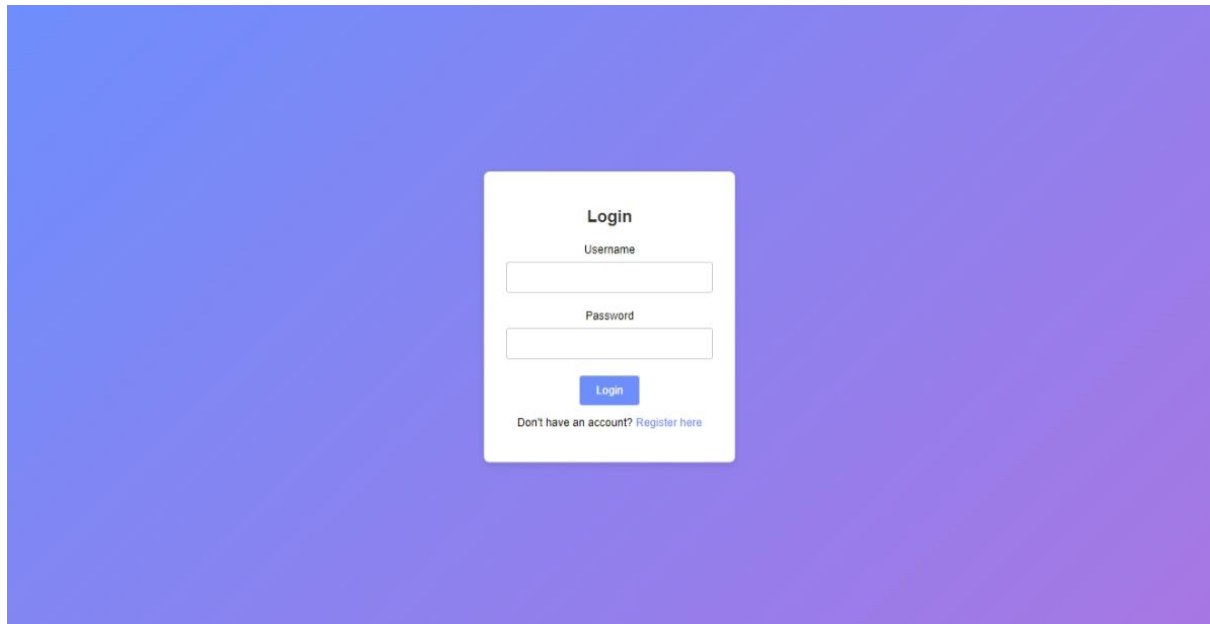
- Now put the main HTML Page of your website (Ex: “index.html”). We have kept static_Website.html as our HTML page



- Then click “**Save changes**” and then there will be a link generated in the static web-hosting option



- And now copy the link and paste it in your browser (**or**) we can just click on the link and it will be redirected to the link.



Login

Username

Password

[Login](#)

Don't have an account? [Register here](#)