

Hosting a Basic Static Website on AWS S3

Introduction

Amazon Simple Storage Service (Amazon S3) allows you to store files securely and serve them over the internet. One powerful feature of S3 is **Static Website Hosting**, which lets you host simple websites made of HTML, CSS, and JavaScript without managing servers.

In this document, we will create a **basic static website** using two HTML files (index.html and error.html) and host it publicly using AWS S3.

What Is a Static Website?

A **static website** consists of fixed files (HTML, CSS, JS) that are served directly to the browser.

Characteristics:

- No server-side processing (no PHP, Node.js, Python, etc.)
- Content does not change dynamically
- Fast, lightweight, and low cost

Why Use AWS S3 for Static Website Hosting?

Using S3 for static websites is popular because:

- **Serverless** – No EC2 or server management
- **Low cost** – Pay only for storage and bandwidth
- **Highly available & durable**
- **Scales automatically**
- Easy integration with **CloudFront, Route 53, HTTPS**

Architecture Overview

User Browser

S3 Static Website Endpoint

index.html or error.html

No backend server is involved.

Step 1: Prepare Website Files

1. Create Project Folder

On your local computer, create a folder named:

my-website

2. Create index.html

This is the **homepage**.

```
<!DOCTYPE html>

<html>
  <head>
    <title>My Static Website</title>
  </head>
  <body>
    <h1>Welcome to My Website</h1>
    <p>This website is hosted on AWS S3.</p>
  </body>
</html>
```

3. Create error.html

This page appears when a user visits a wrong URL.

```
<!DOCTYPE html>

<html>
  <head>
    <title>Error</title>
  </head>
  <body>
    <h1>Oops! Page not found.</h1>
    <p>The page you are looking for does not exist.</p>
```

```
</body>
```

```
</html>
```

The screenshot shows two instances of Visual Studio Code side-by-side, illustrating the process of creating a static website.

Top Window (Initial State):

- Title Bar:** my-website
- Start Panel:** Contains links for "New File...", "Open File...", "Open Folder...", "Clone Git Repository...", "Connect to...", and "Generate New Workspace...".
- Walkthroughs:** Displays "Learn the Fundamentals" and "Get Started With GitLens Updated".
- Recent:** Lists recent files including "Desktop", "thattukada-game", "MathProblems", "no-title", "un-titled", and "More...".
- Bottom Status Bar:** Shows "Show welcome page on startup" checked.

Bottom Window (After Creation):

- Title Bar:** my-website
- Explorer:** Shows a file tree with "index.html" selected.
- Editor:** Displays the content of index.html:

```
<!DOCTYPE html>
<html>
<head>
<title>My Static Website</title>
</head>
<body>
<h1>Welcome to My Website</h1>
<p>This website is hosted on AWS S3.</p>
</body>
</html>
```
- Bottom Status Bar:** Shows "Ln 11, Col 1" and other standard status bar information.

```
<!DOCTYPE html>
<html>
<head>
<title>Error</title>
</head>
<body>
<h1>Oops! Page not found.</h1>
<p>The page you are looking for does not exist.</p>
</body>
</html>
```

The screenshot shows a code editor interface with a dark theme. On the left, there's a tree view of files under 'MY-WEBSITE': 'error.html' and 'index.html'. The main area displays the content of 'error.html'. The code is as follows:

```
<!DOCTYPE html>
<html>
<head>
<title>Error</title>
</head>
<body>
<h1>Oops! Page not found.</h1>
<p>The page you are looking for does not exist.</p>
</body>
</html>
```

At the bottom of the editor, status bar text includes: Ln 11, Col 1, Spaces: 4, UTF-8, CRLF, HTML, Finish Setup, Prettier.

Step 2: Create an S3 Bucket

1. Sign in to the **AWS Management Console**
2. Open **Amazon S3**
3. Click **Create bucket**
4. Enter a **unique bucket name**
Example:
 - 5. my-static-website-anshad-05
 - 6. Choose a region (example: us-east-1)
 - 7. **Uncheck:**
 - 8. Block all public access
 - 9. Acknowledge the warning
 - 10. Click **Create bucket**

S | Storage | [Alt+S] | Search | United States (N. Virginia) | MhdAnshad (1295-8554-0940) | MhdAnshad

Amazon S3

Store and retrieve any amount of data from anywhere

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.

Create a bucket

Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.

Create bucket

Pricing

With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket.

Estimate your monthly bill using the [AWS Simple Monthly Calculator](#).

[View pricing details](#)

How it works



Resources

[User guide](#)

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

S | AWS | Storage | [Alt+S] | Search | United States (N. Virginia) | MhdAnshad (1295-8554-0940) | MhdAnshad

Amazon S3 > Buckets > Create bucket

General configuration

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

Bucket type: General purpose

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: 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: my-static-website-123

Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). [Learn more](#)

Copy settings from existing bucket - optional

Only the bucket settings in the following configuration are copied.

Choose bucket

Format: s3://bucket/prefix

Object Ownership

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.

Object Ownership:

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.

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

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](#)

General purpose buckets [All AWS Regions](#) **Directory buckets**

General purpose buckets (1) [Info](#) [Delete](#) [Create bucket](#)

Buckets are containers for data stored in S3.

Name	AWS Region	Creation date
my-static-website-anshad-05	US East (N. Virginia) us-east-1	January 15, 2026, 15:35:38 (UTC+05:30)

Account snapshot [Info](#)
Updated daily
[View dashboard](#)
Storage Lens provides visibility into storage usage and activity trends.

External access summary [Info](#)
Updated daily
External access findings help you identify bucket permissions that allow public access or access from other AWS accounts.

Step 3: Enable Static Website Hosting

1. Open the created bucket
2. Go to the **Properties** tab
3. Scroll to **Static website hosting**
4. Click **Edit**
5. Select **Enable**
6. Enter:

7. Index document: index.html
8. Error document: error.html
9. Save changes
10. Copy the **Bucket website endpoint** (used later for testing)

my-static-website-anshad-05 Info

Objects | Metadata | **Properties** | Permissions | Metrics | Management | Access Points

Bucket overview

AWS Region: US East (N. Virginia) us-east-1 | Amazon Resource Name (ARN): arn:aws:s3:::my-static-website-anshad-05 | Creation date: January 15, 2026, 15:35:38 (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 Edit

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 SDK, or the Amazon S3 REST API. [Learn more ↗](#)

Bucket ABAC Edit

Attribute-based access control (ABAC) is an authorization strategy that defines permissions based on attributes. With ABAC, you can attach tags to your general purpose buckets and AWS Identity and Access Management (IAM) entities (users or roles), then scale access to objects in your S3 general purpose buckets using tag-based policies. [Learn more ↗](#)

CloudShell | Feedback | Console Mobile App | © 2026, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences

Edit static website hosting Info

Static website hosting Edit

Use this bucket to host a website or redirect requests. [Learn more ↗](#)

Static website hosting

Disable Enable

Cancel **Save changes**

CloudShell | Feedback | Console Mobile App | © 2026, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences

Static website hosting

Use this bucket to host a website or redirect requests. [Learn more ↗](#)

Static website hosting

Disable
 Enable

Hosting type

Host a static website
 Use the bucket endpoint as the web address. [Learn more ↗](#)

Redirect requests for an object
 Redirect requests to another bucket or domain. [Learn more ↗](#)

ⓘ For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see [Using Amazon S3 Block Public Access](#) ↗

Index document

Specify the home or default page of the website.

index.html

Error document - optional

This is returned when an error occurs.

error.html

Redirection rules - optional

Redirection rules, written in JSON, automatically redirect webpage requests for specific content. [Learn more ↗](#)

1 |

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences MhdAnshad (1295-8554-0940) ▾ MhdAnshad

ⓘ Successfully edited static website hosting.

Requester pays

When enabled, the requester pays for requests and data transfer costs, and anonymous access to this bucket is disabled. [Learn more ↗](#)

Requester pays

Disabled

Static website hosting

Use this bucket to host a website or redirect requests. [Learn more ↗](#)

ⓘ We recommend using AWS Amplify Hosting for static website hosting
 Deploy a fast, secure, and reliable website quickly with AWS Amplify Hosting. Learn more about [Amplify Hosting](#) ↗ or [View your existing Amplify apps](#) ↗

[Create Amplify app ↗](#)

S3 static website hosting

Enabled

Hosting type

Bucket hosting

Bucket website endpoint

When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. [Learn more ↗](#)

HTTP://<http://my-static-website-anshad-05.s3-website-us-east-1.amazonaws.com> ↗

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences MhdAnshad (1295-8554-0940) ▾ MhdAnshad

Step 4: Upload Website Files

1. Open the **Objects** tab

2. Click **Upload**

3. Add:

- index.html
- error.html

4. Click **Upload**

Screenshot of the AWS S3 console showing the bucket 'my-static-website-anshad-05'. The 'Objects' tab is selected. A message states 'No objects' and 'You don't have any objects in this bucket.' A blue 'Upload' button is visible.

Screenshot of the AWS S3 console showing the 'Upload' page for the bucket 'my-static-website-anshad-05'. The 'Upload' tab is selected. A message says 'Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDKs or Amazon S3 REST API.' A large dashed box allows dragging and dropping files. Below it, a table lists 'Files and folders (2 total, 375.0 B)' with two items: 'error.html' and 'index.html'. The 'Destination' section shows the destination as 's3://my-static-website-anshad-05'. The 'Permissions' section is partially visible at the bottom.

Upload: status

After you navigate away from this page, the following information is no longer available.

Summary

Destination	Succeeded	Failed
s3://my-static-website-anshad-05	2 files, 375.0 B (100.00%)	0 files, 0 B (0%)

Files and folders (2 total, 375.0 B)

Name	Folder	Type	Size	Status	Error
error.html	my-website/	text/html	187.0 B	Succeeded	-
index.html	my-website/	text/html	188.0 B	Succeeded	-

Step 5: Configure Public Access (Bucket Policy)

Why This Step Is Required

By default, S3 blocks public access. To allow anyone on the internet to view your website, we must add a **bucket policy**.

Bucket Policy JSON

Replace my-static-website-123 with **your bucket name**.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "PublicReadGetObject",  
      "Effect": "Allow",  
      "Principal": "*",  
      "Action": "s3:GetObject",  
      "Resource": "arn:aws:s3:::my-static-website-123/*"  
    }  
  ]  
}
```

Apply the Policy

1. Go to **Permissions** tab
2. Open **Bucket policy**
3. Paste the JSON
4. Save changes

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

No policy to display.

Object Ownership

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Policy

```
1 Version: "2012-10-17",
2 Statement: [
3   {
4     Sid: "PublicReadGetObject",
5     Effect: "Allow",
6     Principal: "*",
7     Action: "s3:GetObject",
8     Resource: "arn:aws:s3:::my-static-website-anshad-05/*"
9   }
10 ]
11 ]
12 }
13 }
```

Edit statement

Select a statement

Select an existing statement in the policy or add a new statement.

+ Add new statement

The screenshot shows the AWS S3 Bucket Policy editor. At the top, there's a green success message: "Successfully edited bucket policy." Below it, a note says "Individual Block Public Access settings for this bucket" with an "Off" switch. The main area is titled "Bucket policy" and contains a JSON code block:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "PublicReadGetObject",  
      "Effect": "Allow",  
      "Principal": "*",  
      "Action": "s3:GetObject",  
      "Resource": "arn:aws:s3:::my-static-website-anshad-05/*"  
    }  
  ]  
}
```

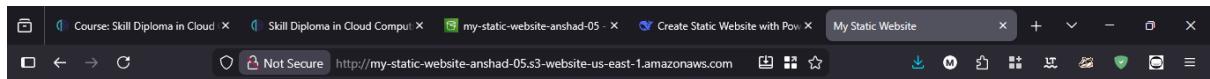
At the bottom right of the code block is a "Copy" button. Below the code block are standard AWS navigation links: CloudShell, Feedback, Console Mobile App, Privacy, Terms, and Cookie preferences.

Step 6: Test the Website

1. Go back to the **Properties** tab
2. Copy the **Static website endpoint URL**
`http://my-static-website-anshad-05.s3-website-us-east-1.amazonaws.com/non-existent-page`
3. Open it in your browser

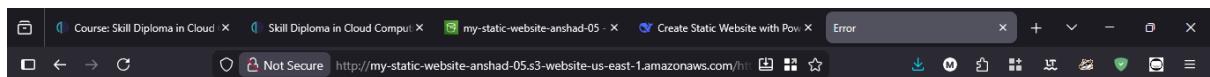
Expected Results:

- Homepage loads → index.html
- Wrong URL (e.g. `http://my-static-website-anshad-05.s3-website-us-east-1.amazonaws.com/non-existent-page`) → error.html



Welcome to My Website

This website is hosted on AWS S3.



Oops! Page not found.

The page you are looking for does not exist.

Common Mistakes & Fixes

Issue	Reason	Fix
403 Forbidden	Bucket policy missing	Add public read policy
Access denied	Public access blocked	Disable “Block all public access”
Blank page	Wrong index name	Use index.html exactly
404 not showing	Error doc missing	Set error.html

Advantages of This Setup

- No server cost
- Easy deployment
- Highly scalable
- Ideal for:
 - Portfolio websites
 - Landing pages
 - Documentation sites
 - Resume websites

Limitations

- No backend logic
- No database support
- Only static content

(Backend can be added later using API Gateway + Lambda if needed.)

Conclusion

Hosting a static website on AWS S3 is one of the **simplest and most cost-effective** ways to publish content on the internet. By enabling static website hosting, uploading HTML files, and configuring public access, you can make your website live within minutes—without managing any servers.