

[nextwork.org](http://nextwork.org)

# Host a Website on Amazon S3

N

Nigel Chitapi

**EVERYONE SHOULD  
BE IN A JOB THEY  
LOVE**

NEXTWORK

Building the best online learning experience to switch careers and upskill.  
Starting with AWS certifications.

Enter your email

# Introducing Today's Project!

## Project overview

In this project, I will demonstrate how to use S3 to host a static website. I am doing this project to learn about AWS and cloud services and how they can be used to store objects in the cloud AND even host websites (how does that work?!)

## Tools and concepts

Services I used were Amazon S3. Key concepts I learnt include Bucket Policies, uploading static website files, index.html, bucket endpoint URLs and ACLs and how the control access to bucket objects

## Time, challenges, and wins

This project took me approximately 1hr 30 mins, including demo time, quiz time and secret mission time. This is definitely archivable in less than 15 mins. The most challenging part was resolving the 403 Forbidden error. It was most rewarding to see the webpage load and be live and public to the world.

# How I Set Up an S3 Bucket

## What I did in this step

In this step, I will create an S3 bucket, because i need to create a storage space for the website files.

## How long it took to create the bucket

Creating an S3 bucket took me less than 5 minutes. I had to go over ACL and Bucket Versioning concepts.

## Region selection

The Region I picked for my S3 bucket was Oregen because it is the region that is closest to me. It is best practise to pick the region close to you because it reduces latency and costs, although this is not important for this project.

## Understanding bucket name uniqueness

S3 bucket names are globally unique! This means no two AWS S3 buckets in the entire world can have the same name. they have to be completely unique



The screenshot shows the AWS S3 Buckets page. At the top, there's a green success message: "Successfully created bucket 'nextwork-website-project-tawanda'. To upload files and folders, or to configure additional bucket settings, choose View details." Below this, there are two tabs: "General purpose buckets" (selected) and "All AWS Regions". A sub-header "Buckets" is shown. On the left, there's a "Create bucket" button. The main area displays a table of buckets:

Name	AWS Region	Creation date
nextwork-website-project-tawanda	US West (Oregon) us-west-2	December 23, 2025, 03:53:52 (UTC-08:00)

On the right, there are two informational boxes: "Account snapshot" (updated daily) and "External access summary - new" (updated daily).

# Upload Website Files to S3

## What I did in this step

In this step, I will upload the website files into my S3 bucket. This is important because without any files there is no website to host.

## Files I uploaded

I uploaded two files to my S3 bucket - they were an index.html file. this determine the structure (i.e what goes into your websites) and a folder with images that will go onto the website itself.

## How the files work together

Both files are necessary for this project as index.html determines the structure but the structure alone does not illustrates the contents of the website. (i.e if index.html say insert image here,) you might not have the image to show and so you need to supply the images and various other assets that the website needs to display.



The screenshot shows the Amazon S3 console interface. The top navigation bar includes 'Amazon S3', 'Buckets', and the specific bucket name 'nextwork-website-project-tawanda'. Below the navigation is a toolbar with actions like 'Copy S3 URI', 'Copy URL', 'Download', 'Open L...', 'Delete', 'Actions', 'Create folder', and 'Upload'. A message about object permissions is displayed. The main area is titled 'Objects (2)' and lists two items: 'index.html' (an HTML file last modified on December 23, 2025) and a folder named 'NextWork - Everyone should be in a job they love\_files/'. The table headers for the objects are Name, Type, Last modified, Size, and Storage class.

Name	Type	Last modified	Size	Storage class
index.html	html	December 23, 2025, 04:04:01 (UTC-08:00)	58.8 KB	Standard
NextWork - Everyone should be in a job they love_files/	Folder	-	-	-

# Static Website Hosting on S3

## What I did in this step

In this step, I will make the website available to the world. This is called Static Website Hosting. This is important because our website files will stay as just files and not turn into websites if I do not do this step.

## Understanding website hosting

Website hosting means putting our websites files onto a web server, which is a computer designed to turn the files into a website page that people can visit.

## How I enabled website hosting

To enable website hosting with my S3 bucket, I went into the properties tab of my bucket and enables static website hosting and I also labelled the the index.html as the file that i would like to host

## Access Control Lists (ACLs)

An ACL is a way to configure permission settings in a bucket. I enabled ACLs so that i can control access to the website files later. There was a pop-up mentioning that AWS recommends disabling ACLs but we will keep it enabled to learn abot ACLs and compare them with bucket policies later.

[Edit static website hosting](#) Info**Static website hosting**

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

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](#)

i 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.

**Error document - *optional***

This is returned when an error occurs.

# Bucket Endpoints

## Understanding bucket endpoint URLs

Once static website is enabled, S3 produces a bucket endpoint URL, which is a url that takes you to the website that you are hosting.

## What I saw when I tested the endpoint

When I first visited the bucket endpoint URL, I saw an error. 403 forbidden error. The reason for this error was because objects in a bucket are private by default even though I switched off block all public access. the website files themselves are still locked. I need to make their public access permissions so that they are visible to the public.



# Success!

## What I did in this step

In this step, I will go into the files that i uploaded into my bucket and make them public, because this will enable the content of the website to be visible.

## How I resolved the 403 error

To resolve this 403 Forbidden error, I navigated to the objects panel on my S3 bucket and check off both the index.html file and the assets folder and then selected actions and chose make public using ACLs. This then made the files publicly accessible to the world.

N

**Nigel Chitapi**  
NextWork Student

[nextwork.org](http://nextwork.org)

The graphic features a dark background with a grid of circular animal photos at the top. The animals include an owl, a kangaroo, a zebra, a giraffe, a cat, an owl, a penguin, a dog, a pinecone, and a black dog. To the right of the grid is a dark circular button with the text "You?". Above the animals, the text "EVERYONE SHOULD BE IN A JOB THEY LOVE" is displayed in large, bold, white and blue letters. The "NEXTWORK" logo is in the top left corner. Below the main title, the text "Building the best online learning experience to switch careers and upskill. Starting with AWS certifications." is written in a smaller font. At the bottom, there is a form with a placeholder "Enter your email" and a blue "Join waitlist" button.

EVERYONE SHOULD  
BE IN A JOB THEY  
LOVE

BUILDING THE BEST ONLINE LEARNING EXPERIENCE TO SWITCH CAREERS AND UPSKILL.  
STARTING WITH AWS CERTIFICATIONS.

Enter your email

Join waitlist

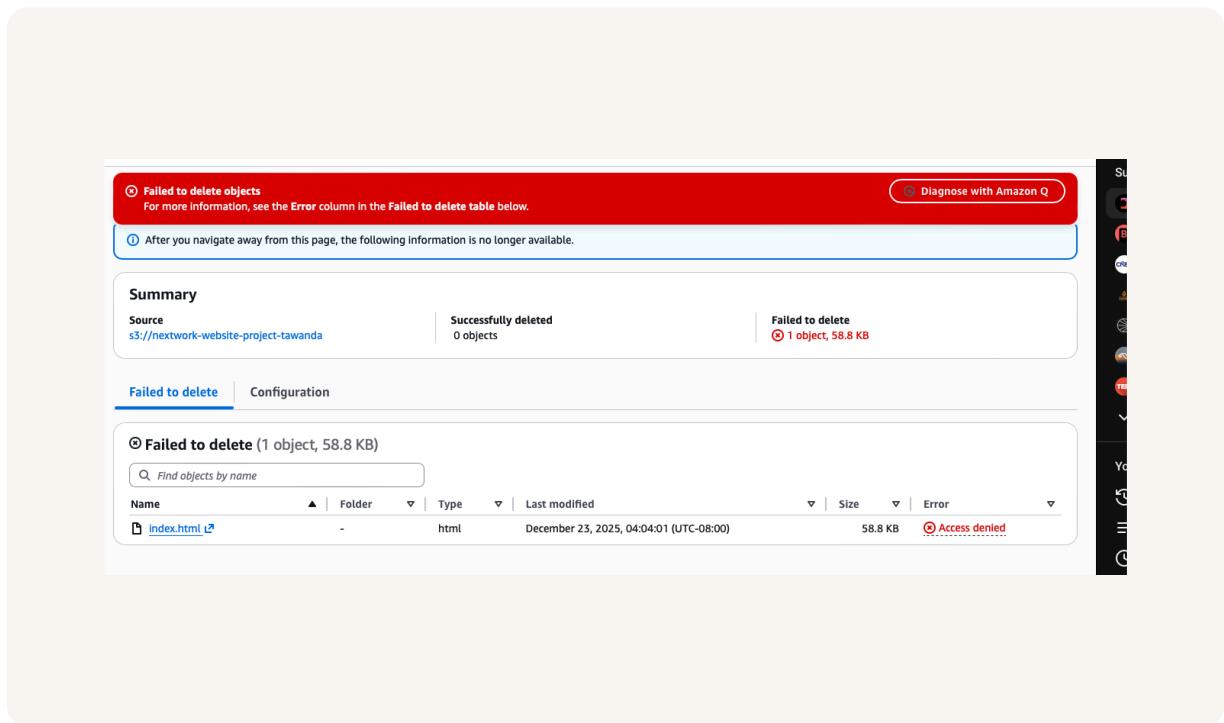
# Bucket Policies

## What I did in this extension

In this project extension I'm about to use Bucket Policies to control access to the bucket files. I'm doing this so that we can stop people from deleting the objects with the buckets.

## Understanding bucket policies

An alternative to ACLs are bucket policies, which are rules that determine who is allowed or not allowed to do something. The benefit of using bucket policies is that you can have even greater control of accesss of the actions that people can or cannot make, while ACLs are useful for controlling public access to individual objects within the bucket.



## What my bucket policy does

My bucket policy denies everyone from deleting the index.html file in the bucket. I tested this by attempting to delete index.html and saw an access denied error, meaning the bucket policy worked effectively. It stopped me from deleting the object i intended to protect.



[nextwork.org](https://nextwork.org)

# The place to learn & showcase your skills

Check out [nextwork.org](https://nextwork.org) for more projects

