

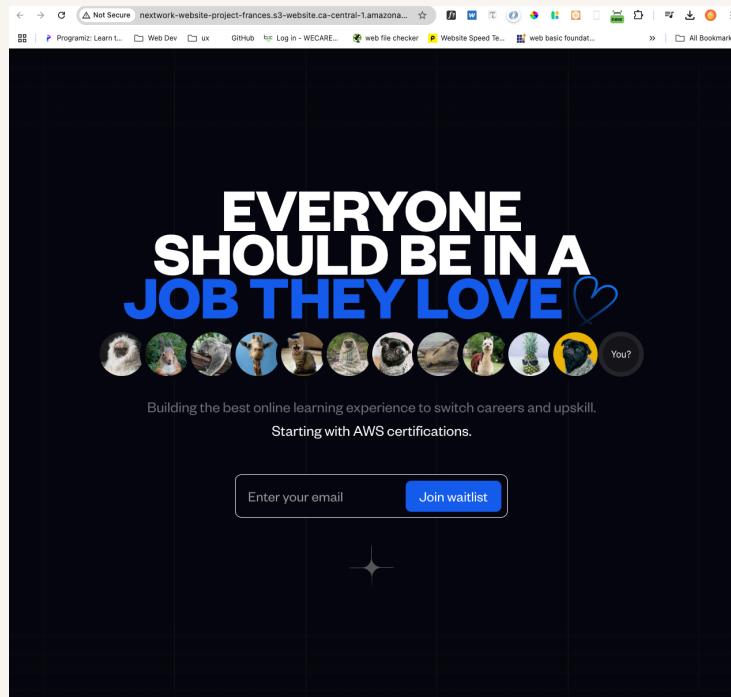


NextWork.org

Host a Website on Amazon S3



francesj7837@gmail.com



Introducing Today's Project!

What is Amazon S3

Amazon S3 is a storage space to upload website files and host them to cloud so that it can be known to public. This is useful because you have full control of who gets to access your bucket.

How I used Amazon S3 in this project

I used it to learn how to create a bucket, upload web files and assets and host them to public.

One thing I didn't expect in this project was...

The error message which I find it useful because I get to learn how to resolve the issue.

This project took me...

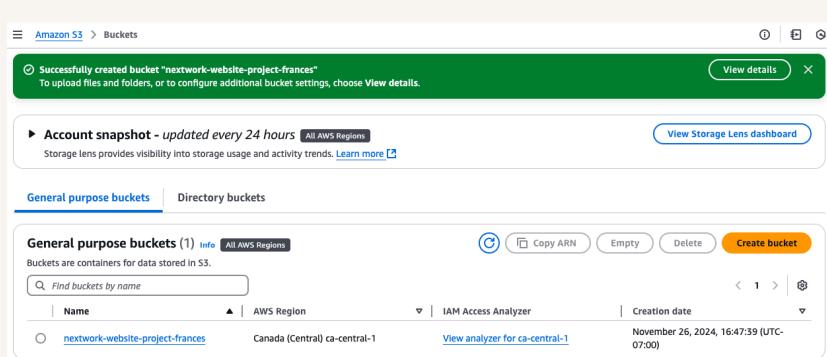
It took me less than 30 mins!

How I Set Up an S3 Bucket

Creating an S3 bucket took me less than 5 mins to create and have a good understanding of the entire set up.

I picked Central Canada as the region because this is where I'm closest to.

S3 bucket names are globally unique this means no other buckets are named after the ones you picked, unless you deleted your own then someone can use it.



Upload Website Files to S3

index.html and image assets

I uploaded two files to my S3 bucket - they were the index.html and a folder named NextWork - Everyone should be in a job they love_files/

Both files are necessary for this project as they contain the asset files that is needed for creating the website.

The screenshot shows the Amazon S3 console interface. The top navigation bar includes 'Amazon S3', 'Buckets', and the specific bucket name 'nextwork-website-project-frances'. Below the navigation is a breadcrumb trail: 'nextwork-website-project-frances' followed by 'Info'. A horizontal menu bar below the breadcrumb trail includes 'Objects' (which is underlined in blue), 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. The main content area is titled 'Objects (2) Info'. It features a toolbar with buttons for 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'. A search bar labeled 'Find objects by prefix' is present. A note below the toolbar states: 'Objects are 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](#)' with a small link icon. The object list table has columns for 'Name', 'Type', 'Last modified', 'Size', and 'Storage class'. The first item, 'index (1).html', is listed as an HTML file last modified on November 26, 2024, at 16:52:41 (UTC-07:00), with a size of 58.8 KB and Standard storage class. The second item, 'NextWork - Everyone should be in a job they love_files/', is listed as a Folder. The 'Upload' button is highlighted with an orange border.

Static Website Hosting on S3

Website hosting means to show my website to the public where anyone who knows about my web URL can view it in their device

To enable website hosting with my S3 bucket, I have to go to Properties tab and select Static website hosting and enable it.

An ACL is managing who can access and what type of files to upload in my bucket.

403 Forbidden

- Code: AccessDenied
- Message: Access Denied
- RequestId: VW5GVSE3X7R5Q8ER
- HostId: SkE9gjL5grkuveosx0gIcmHby3PkCL8sKcQLBiCDNnw6RX9vrKbUMTm8K03Vj+Xz2X/gQ2e4to=

Bucket Endpoints

Once static website is enabled, S3 produces a bucket endpoint URL, which is the link to my website

When I first visited the bucket endpoint URL, I saw the 403 error message. The reason for this error was that the objects that I uploaded are not made for public yet so I have to take action and make them public.

403 Forbidden

- Code: AccessDenied
- Message: Access Denied
- RequestId: VW5GV5E3X7R5Q8ER
- HostId: SkE9gjL5grkuveosx0gIcmHby3PkCL8sKcQKLBiCDNnw6RX9vrKbUMTm8K03Vj+Xz2X/gQ2e4to=

FR

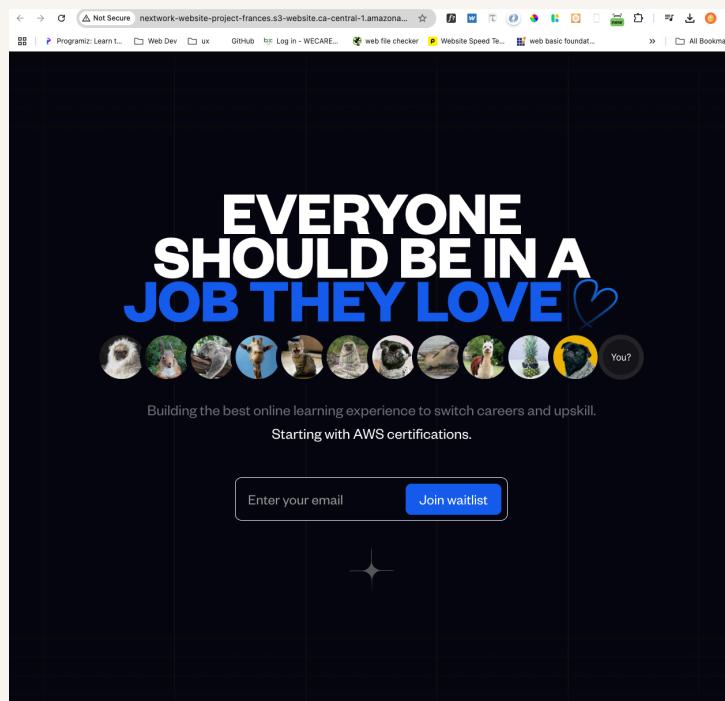
francesj7837@gmail.com

NextWork Student

NextWork.org

Success!

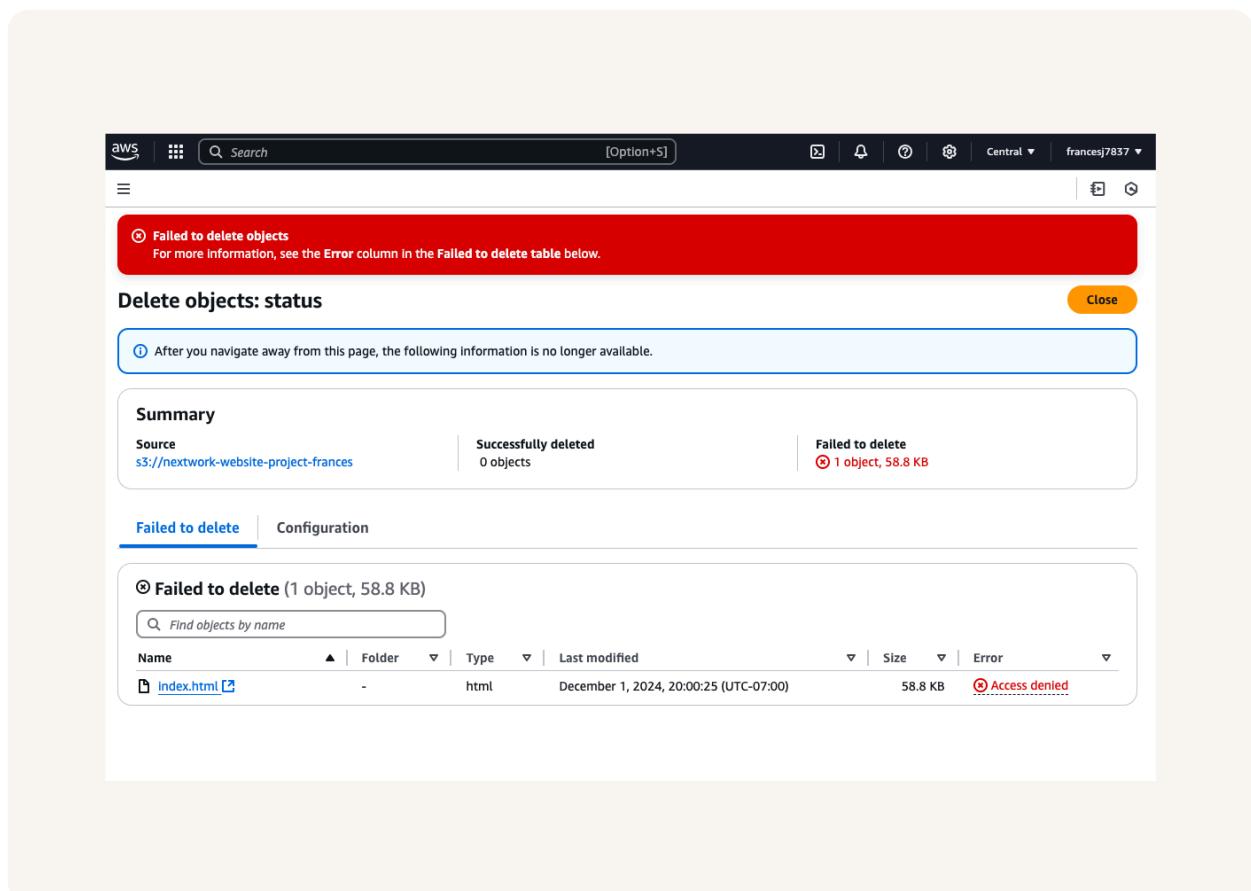
To resolve this connection error, I have to go to Objects tab, select all the files I uploaded, go to Actions and select Make public using ACL.



Bucket Policies

An Alternative to ACLs are bucket policies, which are conditions for managing files inside the bucket. The benefit of using bucket policies is I can control the objects available in the bucket while ACLs are useful for delete, change and more.

My bucket policy that I set the file named index.html not to be deleted was saved. I tested this by trying to delete the file and saw the message that says 'Failed to delete objects'. This means the policy that I created worked.



The screenshot shows the AWS S3 console interface. At the top, there's a red banner with the message: "Failed to delete objects. For more information, see the Error column in the Failed to delete table below." Below this, a modal window titled "Delete objects: status" is open. It contains a message: "After you navigate away from this page, the following information is no longer available." Under the "Summary" section, it shows "Source s3://nextwork-website-project-frances", "Successfully deleted 0 objects", and "Failed to delete 1 object, 58.8 KB". A "Close" button is visible in the top right of the modal. Below the summary, there are two tabs: "Failed to delete" (which is selected) and "Configuration". The "Failed to delete" tab displays a table with one row. The table has columns for Name, Type, Last modified, Size, and Error. The single entry is "index.html" (Type: html, Last modified: December 1, 2024, 20:00:25 (UTC-07:00), Size: 58.8 KB, Error: Access denied).

Name	Type	Last modified	Size	Error
index.html	html	December 1, 2024, 20:00:25 (UTC-07:00)	58.8 KB	Access denied



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

