

How to Setup a Simple Storage Service, S3

In this documentation, we will be discussing the benefits, usage and setup/teardown of the Simple Storage Service, or S3. Let us begin with disclosing what the service S3 is.

S3, stands for ***Simple Storage Service***. *It is a global object storage service. S3 can be used to store and recover any amount of data, at anytime from anywhere utilizing a public internet connection.*

It is a ***global storage service***, that can be accessed anywhere from within AWS platform. The data/files that are stored using S3 stays within the region it was uploaded in, making S3 ***regionally based***. The data does NOT leave that region unless arranged to do so. It is ***regionally resilient***, meaning it can be accessed across all the availability zones, again, within a specific region. Therefore, if any zone goes down, the data is not lost.

It is a public service, where an unlimited amount of data can be stored and reached by a multitude of users.

S3 is *designed to hold large sets of data*. These files could be movies, audio, text, photos data for IoT devices, blog posts or website data. The data can be accessed through GUI's (graphical user interface) CLI (command line) AWS API (application programming interface) and HTTP (hypertext transfer protocol).

Objects are files or data that can be uploaded, stored, accessed and deleted from buckets. ***Buckets*** are the containers in which the objects are deposited.

Objects are made up of several parts, along with information about the data or *metadata*. One part of an Object is a ***key*** which is the file name of the object. If you have the file name, you can easily search to retrieve the second part of the object, which is the ***value***. The ***value*** is the data/content itself. An objects value is its size and that *ranges from zero bytes to 5 terabytes*.

Buckets once created in a region, never leave that region unless made to do so by an authorized user. A bucket is identified by its name or *bucket name*, which must be a unique name identifier in **ALL** the AWS core platform. This makes bucket names **globally unique**. Buckets can hold an **unlimited** amount of data/objects. The objects are stored together, on the same level as other objects in the bucket. This type of storage is a *flat structure*. There are no file types in buckets, everything is an object or a piece of data.

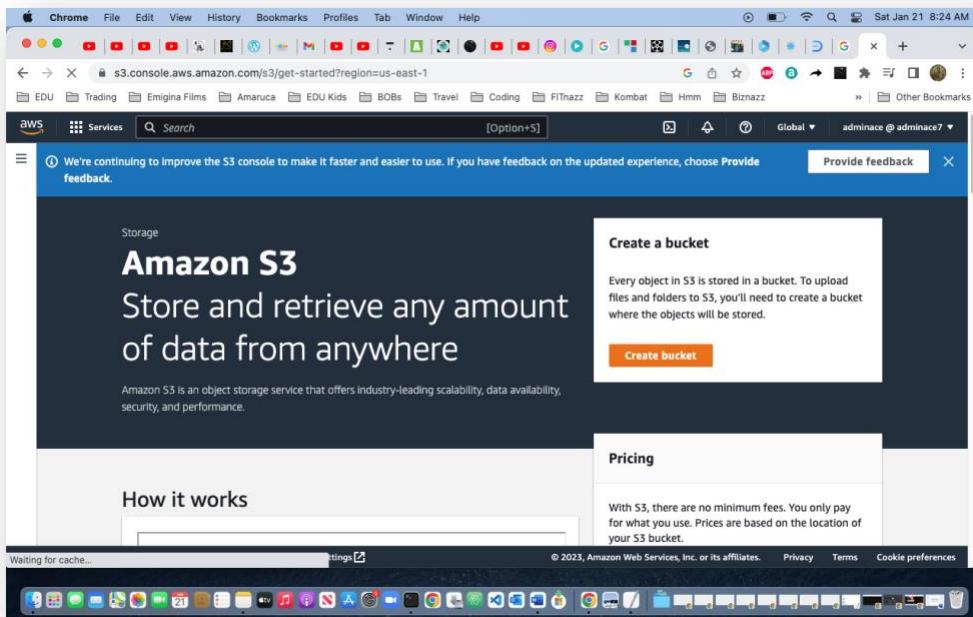
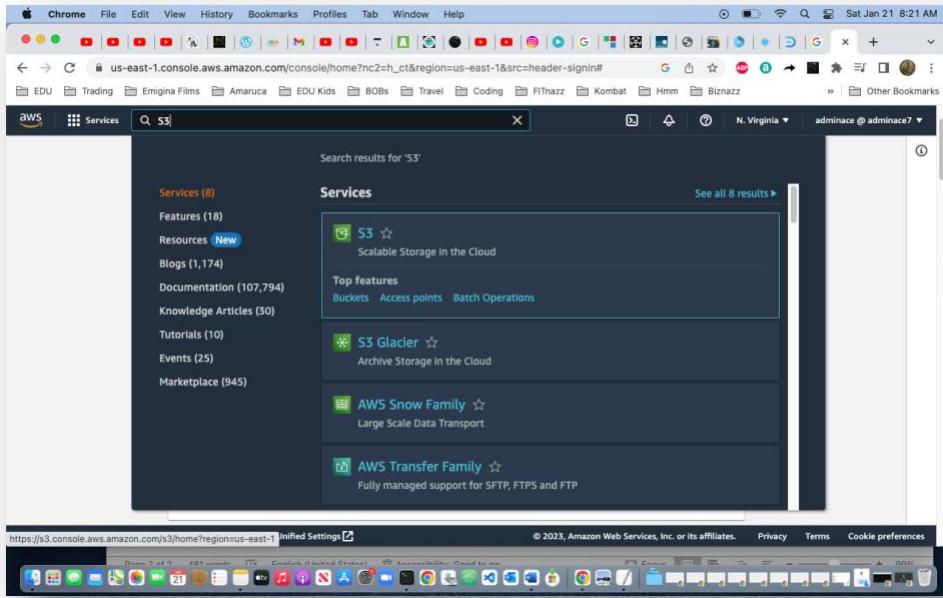
To create folders of objects within a bucket, you have to prefix the key of an object with a slash and then the intended folder name. An example of this is object key cowboy.jpg. To put this object in a folder (folder named west) prefix it with a slash then type west, followed by another slash: /west/cowboy.jpg.

Buckets are the default area where options and permissions(rules) are set in S3.

Now we shall begin the setup process of an S3 bucket with objects, that will operate as a static website able to be reached by HTTP.

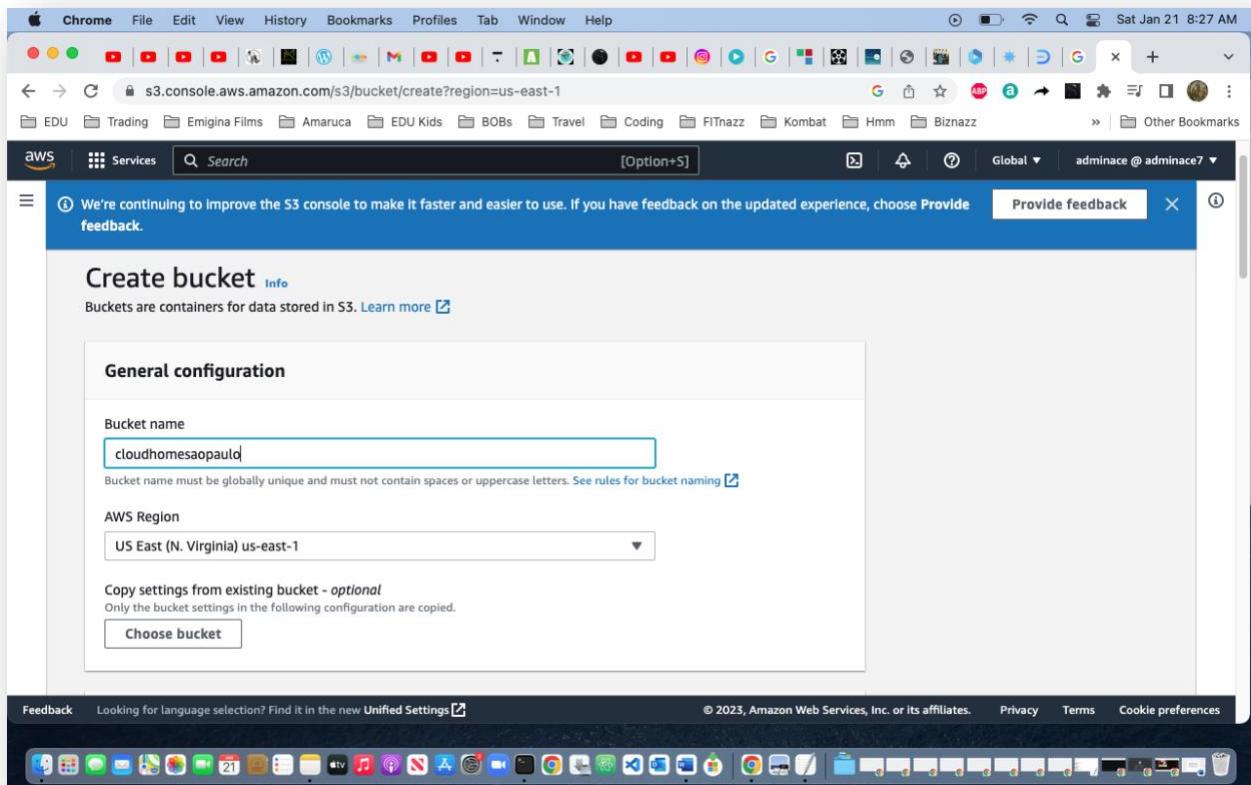
S3 Setup as a Static Website

Go inside your AWS account and type “S3” in the **find services search box**. Select the “S3” option to arrive at the **S3 main page**. Select the “*create bucket*”, button.



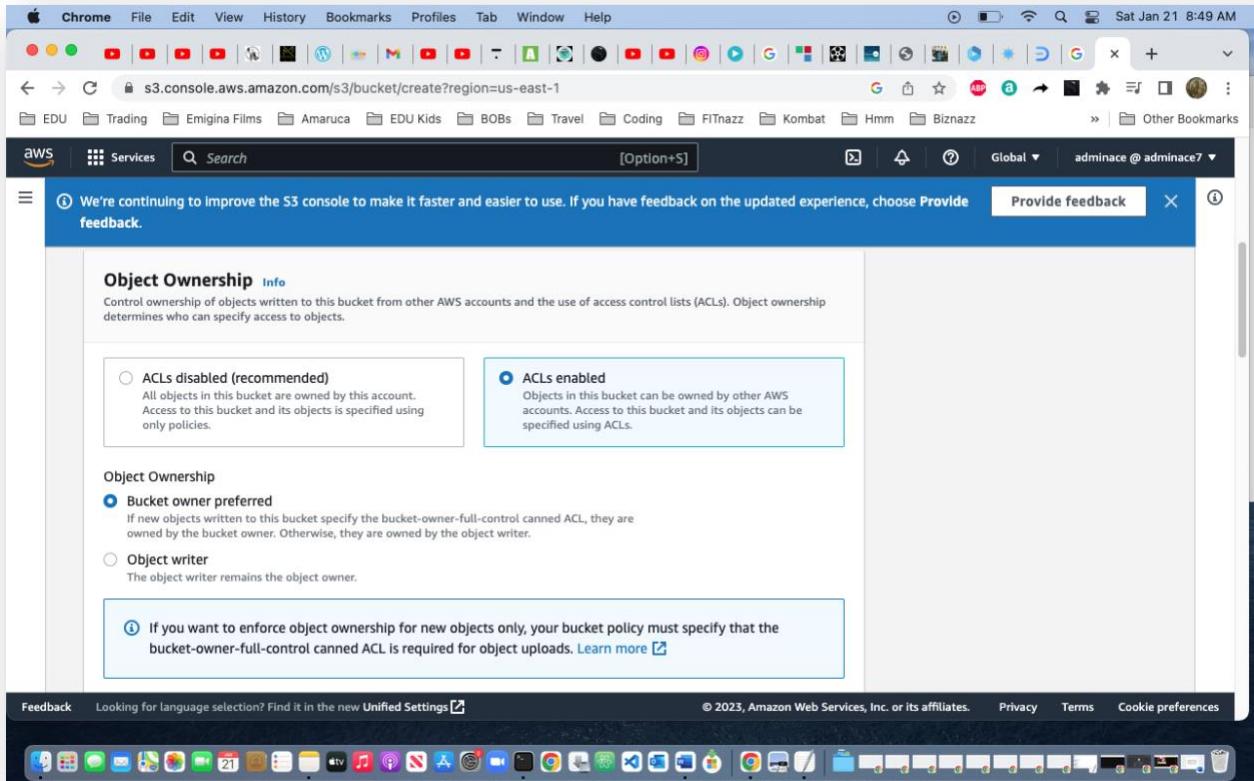
On the **Create Bucket** page, under the *General configuration* section, give your bucket a **Globally Unique**, name. To do so, use these best practices below in coming up with a bucket name:

- Use between 3-63 characters, that are all lower case and can start with a letter or number, except the prefix xn--.
- Permissible characters can be dots(.), and hyphens (-). No underscores.
- No adjacent periods.
- Cannot format the name in the same manner as an IP address.



Next, in the *Object Ownership* section we will grant access to our bucket and objects. We do that through **ACL's or Access Control Lists**. *ACLs allows the user to set rules on which accounts/groups have access to the bucket and objects, as well the type of access permitted.*

We will select the “*ACLs enabled*”, option. This option will be used to allow HTTP traffic access to our bucket, so we can access it through a public internet.



For the *Block Public Access settings for this bucket*, section, we will uncheck the “block all public access”, box. This is ONE of the steps necessary to allow public access over HTTP to our bucket and objects. Then we will acknowledge our consent in allowing access in the warning dialog box below.

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 your bucket is private, you can 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.

Block Public Access settings for this bucket

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.

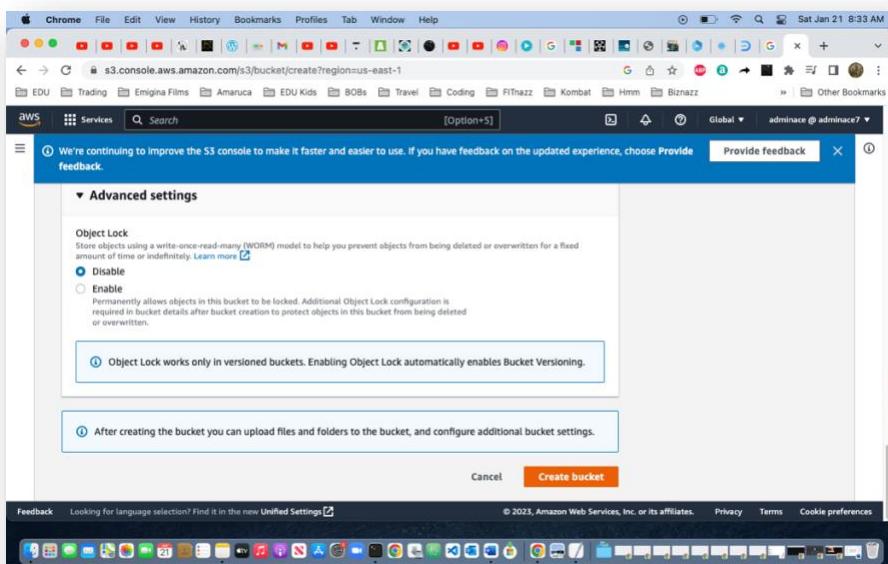
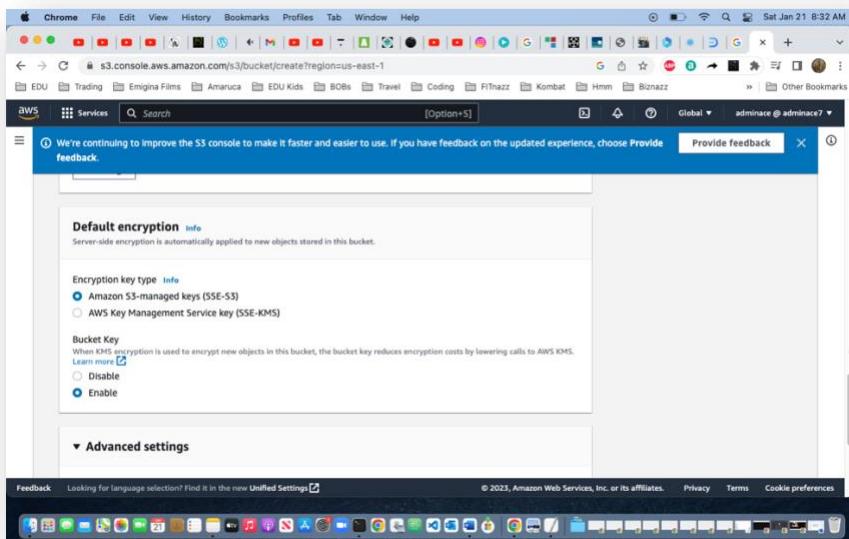
⚠ Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

I acknowledge that the current settings might result in this bucket and the objects within becoming public.

We can skip some sections as we are making a basic bucket with objects. At the *Default encryption* section we select the “*Amazon s3-managed keys*” for *Encryption key type* and the “*enable*”, option under the *Bucket key* subsections.

This section gives AWS authority to handle all encryption of data stored in your S3 bucket.

Then you can scroll to the bottom to select the “*create bucket*”, button, to finish configuring the bucket.

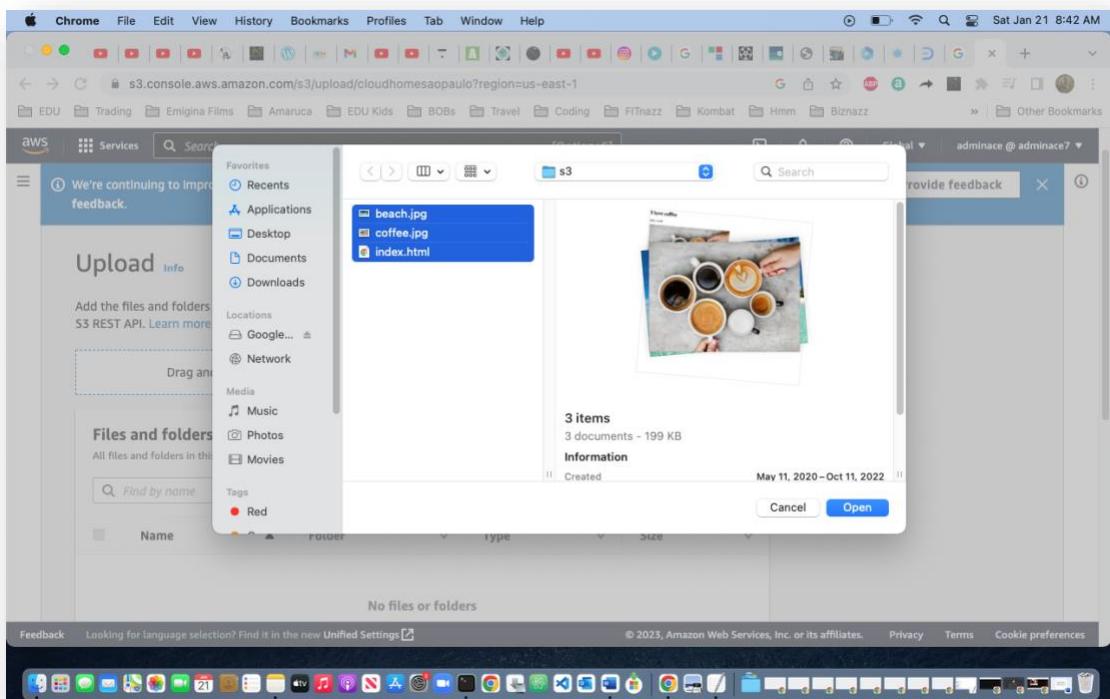
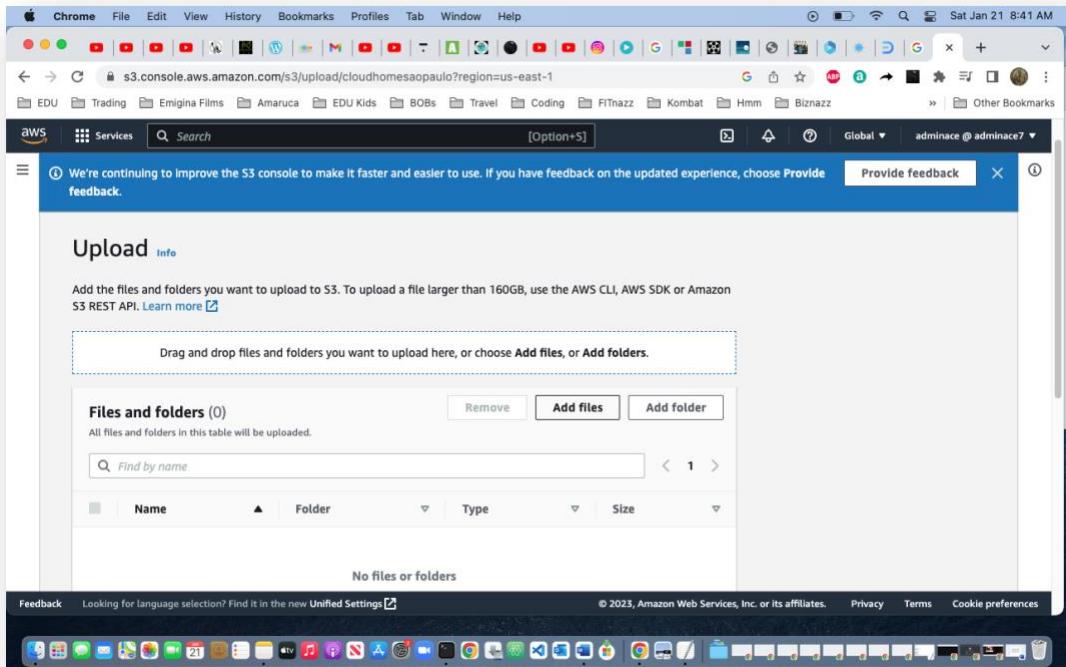


As you can see below, we have our first S3 bucket successfully completed. Now we can begin adding objects to it. Select the “*upload*” button to begin.

The screenshot shows a Chrome browser window with the AWS S3 console at <https://s3.console.aws.amazon.com/s3/buckets?region=us-east-1>. The left sidebar has 'Amazon S3' selected under 'Buckets'. A success message at the top says 'Successfully created bucket "cloudhomesaopaulo"'. Below it, a tutorial message encourages getting started with S3 Intelligent-Tiering. The main area displays a table of buckets, showing one entry: 'cloudhomesaopaulo' in 'US East (N. Virginia) us-east-1' with 'Objects can be public' and 'Creation date' as 'January 21, 2023, 08:33:47 (UTC-05:00)'.

The screenshot shows the same AWS S3 console after navigating to the bucket 'cloudhomesaopaulo'. The left sidebar still shows 'Amazon S3' under 'Buckets'. The main area is titled 'cloudhomesaopaulo' and shows the 'Objects' tab selected. It displays a message about objects being fundamental entities. Below is a table with a single row for 'Upload' with a 'Create folder' button next to it. A search bar at the bottom allows finding objects by prefix.

Here at the ***Upload*** page, we will go to the “*Files and folders*” section and select “*Add files*”. We will select the 3 files from the *S3 folder* downloaded from our Stephane Maarek Udemy course.



Next we scroll down to the “*Permissions*” section to enable our ACLs to allow access to our newly added objects.

The screenshot shows the AWS S3 console in a web browser. The URL is `s3.console.aws.amazon.com/s3/upload/cloudhomesaopaulo?region=us-east-1`. The interface includes a feedback message about improving the console, a search bar, and a navigation bar with tabs like Global and adminace @ adminace7. Below the navigation is a large central area for file uploads. A message says "Drag and drop files and folders you want to upload here, or choose Add files, or Add folders." A table lists three files: beach.jpg (image/jpeg, 85.8 KB), coffee.jpg (image/jpeg, 108.4 KB), and index.html (text/html, 200.0 B). At the bottom, there's a "Destination" section with a dropdown set to `s3://cloudhomesaopaulo`, and sections for "Destination details" and "Permissions". The status bar at the bottom shows the date and time: Sat Jan 21 8:42 AM.

This screenshot continues from the previous one, showing the "Destination" and "Permissions" sections expanded. The "Destination" dropdown is still set to `s3://cloudhomesaopaulo`. The "Permissions" section is expanded, showing the option to "Grant public access and access to other AWS accounts." The "Properties" section is also visible. At the bottom right of the main content area, there is a small preview window showing the contents of the S3 bucket. The browser status bar at the bottom indicates the date and time: Sat Jan 21 8:42 AM.

At the “*Access control list*”, section we will select the “choose from predefined ACLs”, under *Access control list* and “grant public read access” under the *Predefined ACLs*. Once this is done we can scroll to the bottom of the page, to select the “upload” button. The next will display a green banner success message for the uploads.

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback.

Access control list (ACL)

Grant basic read/write permissions to other AWS accounts. Learn more [\[?\]](#)

AWS recommends using S3 bucket policies or IAM policies for access control. Learn more [\[?\]](#)

Access control list (ACL)

- Choose from predefined ACLs
- Specify individual ACL permissions

Predefined ACLs

- Private (recommended)
Only the object owner will have read and write access.
- Grant public-read access
Anyone in the world will be able to access the specified objects. The object owner will have read and write access. Learn more [\[?\]](#)

Granting public-read access is not recommended
Anyone in the world will be able to access the specified objects. Learn more [\[?\]](#)

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback.

Upload succeeded
View details below.

Upload: status

The information below will no longer be available after you navigate away from this page.

| Summary | | |
|--|---|---|
| Destination s3://clouhomesaopaulo | Succeeded [?] 3 files, 194.4 KB (100.0%) | Failed [?] 0 files, 0 B (0%) |

Files and folders | Configuration

As shown below, we successfully upload our objects. On this page, we select the “close” button to arrive at our objects page.

The screenshot shows a Chrome browser window with the AWS S3 console URL: s3.console.aws.amazon.com/s3/upload/cloudhomesaopaulo?region=us-east-1. The interface includes a top navigation bar with tabs like File, Edit, View, History, Bookmarks, Profiles, Tab, Window, Help, and a search bar. Below the navigation is a toolbar with various icons. A blue header bar displays a message: "We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback." A green success message box says "Upload succeeded" and "View details below." Below this, there are two tabs: "Files and folders" (selected) and "Configuration". Under "Files and folders", a table lists three files: beach.jpg (image/jpeg, 85.8 KB, Succeeded), coffee.jpg (image/jpeg, 108.4 KB, Succeeded), and index.html (text/html, 200.0 B, Succeeded). At the bottom of the page, there's a feedback link, a copyright notice (© 2023, Amazon Web Services, Inc. or its affiliates.), and links for Privacy and Terms. A "Close" button is located in the top right corner of the main content area.

This screenshot shows the same AWS S3 console page after selecting the "Close" button from the previous screenshot. The main content area now displays a summary titled "Upload: status". It contains a message: "The information below will no longer be available after you navigate away from this page." Below this is a "Summary" section with a table:

| Destination | Succeeded | Failed |
|------------------------|-----------------------------|-------------------|
| s3://cloudhomesaopaulo | 3 files, 194.4 KB (100.00%) | 0 files, 0 B (0%) |

At the bottom of the page, there are tabs for "Files and folders" and "Configuration", along with standard footer links for Feedback, Unified Settings, Privacy, Terms, and Cookie preferences. A "Close" button is also present in the top right corner of the summary box.

At the objects page, we can see the 3 files we have uploaded into the bucket. It is at this stage we will begin making each object public through our ACLs.

Check the box next to the hypertext *index.html* object. Then go to the “Actions” button. Press it, then scroll to select the “*Make public using ACL*”, option.

The screenshot shows the AWS S3 console in a web browser. The URL is `s3.console.aws.amazon.com`. The page displays three objects: `beach.jpg`, `coffee.jpg`, and `index.html`. The `index.html` file is selected, indicated by a checked checkbox in the left column. An "Actions" dropdown menu is open over the selected file, with the "Make public using ACL" option highlighted. Other options visible in the dropdown include "Rename object", "Edit storage class", "Edit server-side encryption", "Edit metadata", and "Edit tags". The browser's status bar shows the date and time as "Sat Jan 21 8:52 AM".

| Name | Type | Last modified |
|-------------------------|------|--|
| <code>beach.jpg</code> | jpg | January 21, 2023, 08:51:58 (UTC-05:00) |
| <code>coffee.jpg</code> | jpg | January 21, 2023, 08:51:58 (UTC-05:00) |
| <code>index.html</code> | html | January 21, 2023, 08:51:59 (UTC-05:00) |

On the ***Make Public*** page, we individually make each object public, one at a time. Go to the ***Specified Objects*** section and type in the key or file name *index.html*, the object we previously selected from the last page. Then select the “*make public*” button at the bottom.

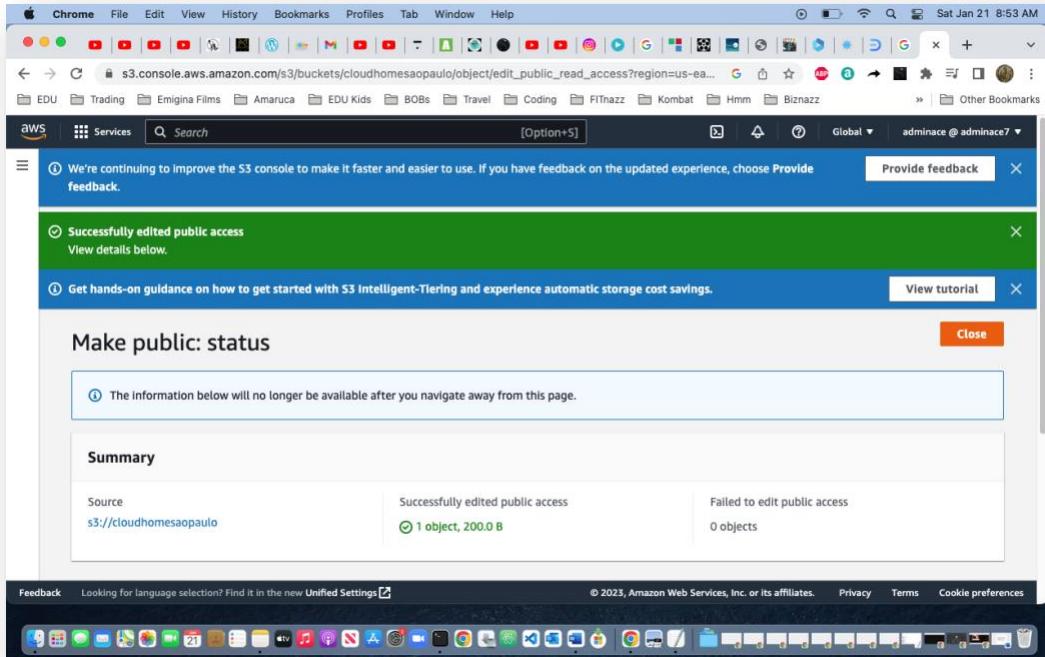
The screenshot shows the AWS S3 console in a Chrome browser. The URL is https://s3.console.aws.amazon.com/s3/buckets/cloudhomesaopaulo/object/edit_public_read_access?region=us-east-1. The page title is "Make public". In the "Specified objects" section, the search bar contains "index.html". A table lists one object:

| Name | Type | Last modified | Size |
|------------|------|--|---------|
| index.html | html | January 21, 2023, 08:51:59 (UTC-05:00) | 200.0 B |

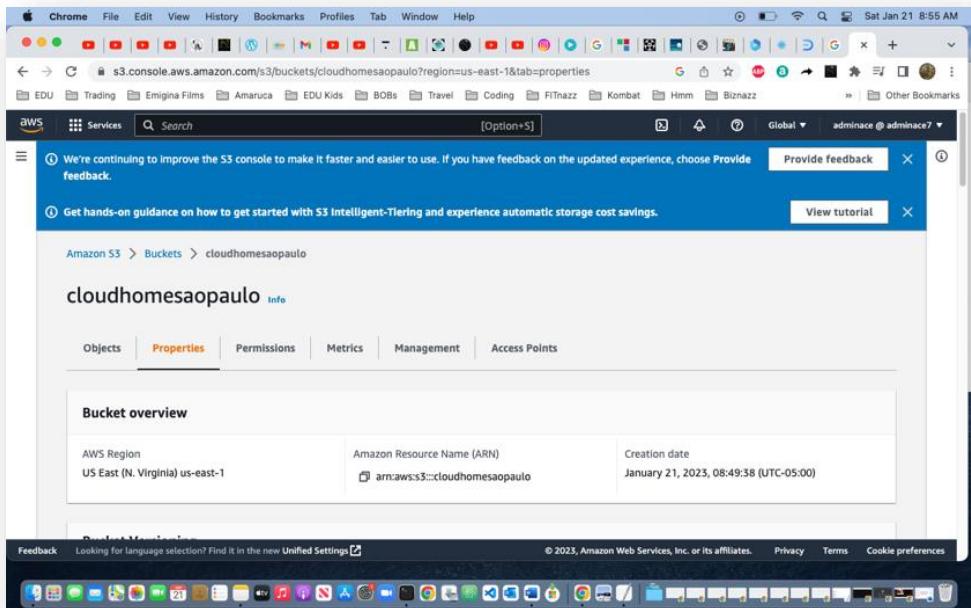
At the bottom right of the table is a red "Make public" button.

This screenshot is identical to the one above, but the "Make public" button at the bottom right of the table has been highlighted with a red rectangle.

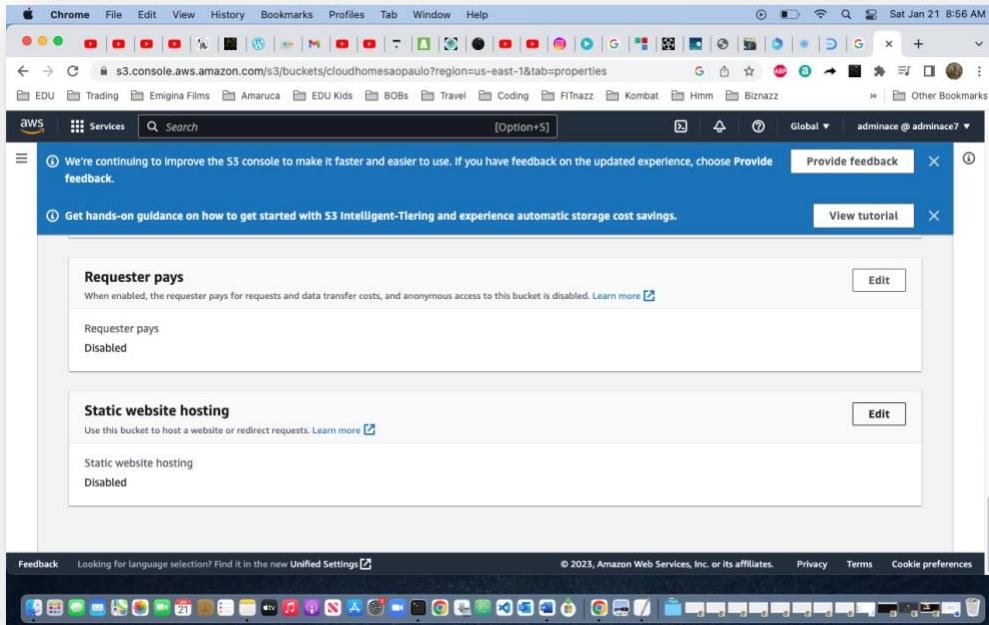
As seen below we successfully made our first object public. Click the “*close*” button to go back to the objects page. We will now repeat this same process to make the last two objects public.



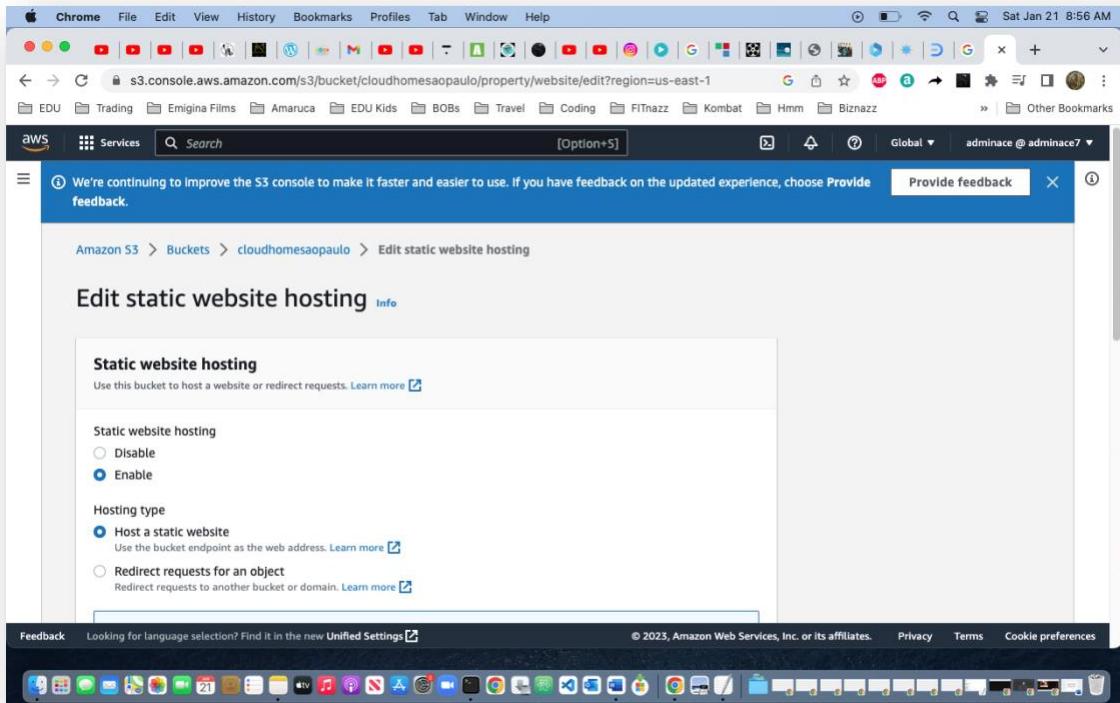
After making all our objects public, we will now enable our static website feature in order to view our objects over the internet. On our objects page, select the *Properties* tab.



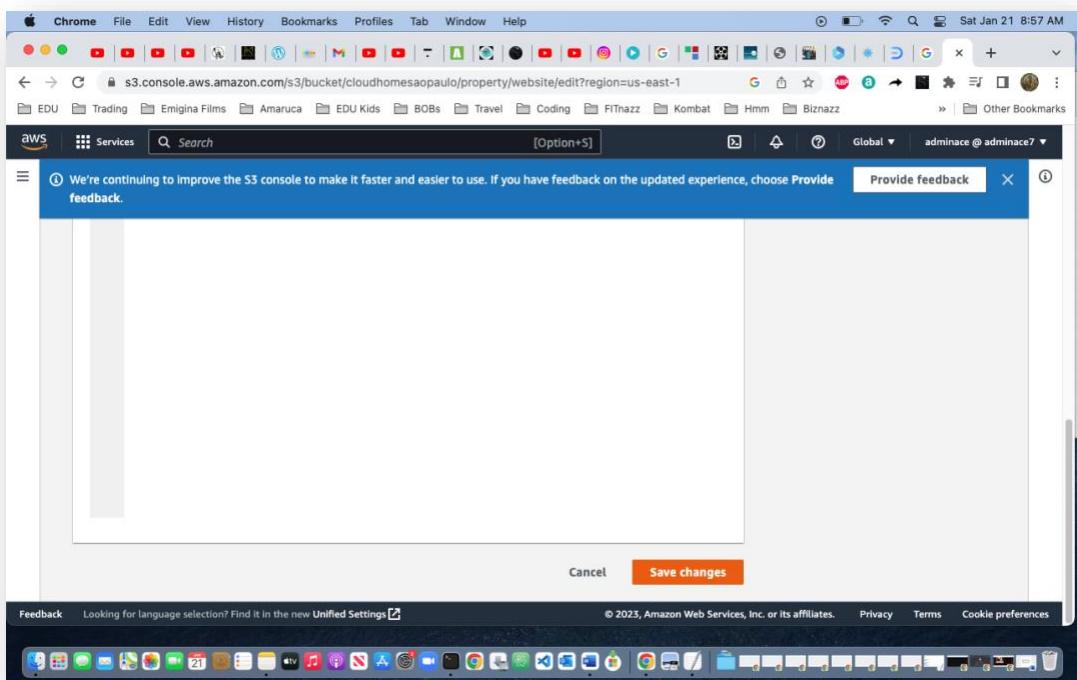
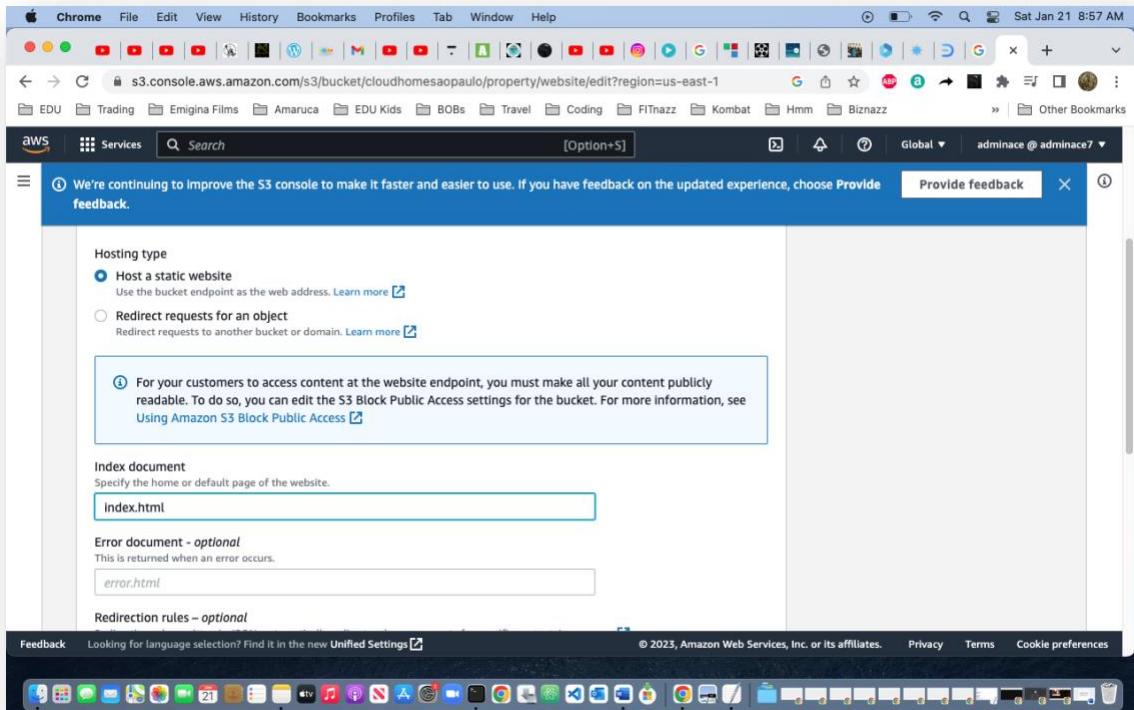
Scroll down the ***Properties*** page until you arrive at the “*static website hosting*” section. Click the edit button to start setup.



In this section, we will select “*enable*”, static web hosting and “*host a static website*” for hosting type.



To direct traffic to our home page, we will type, “*index.html*” in the input field under the *index document* subsection. We then scroll to the bottom of the page to select the “*save changes*” button.

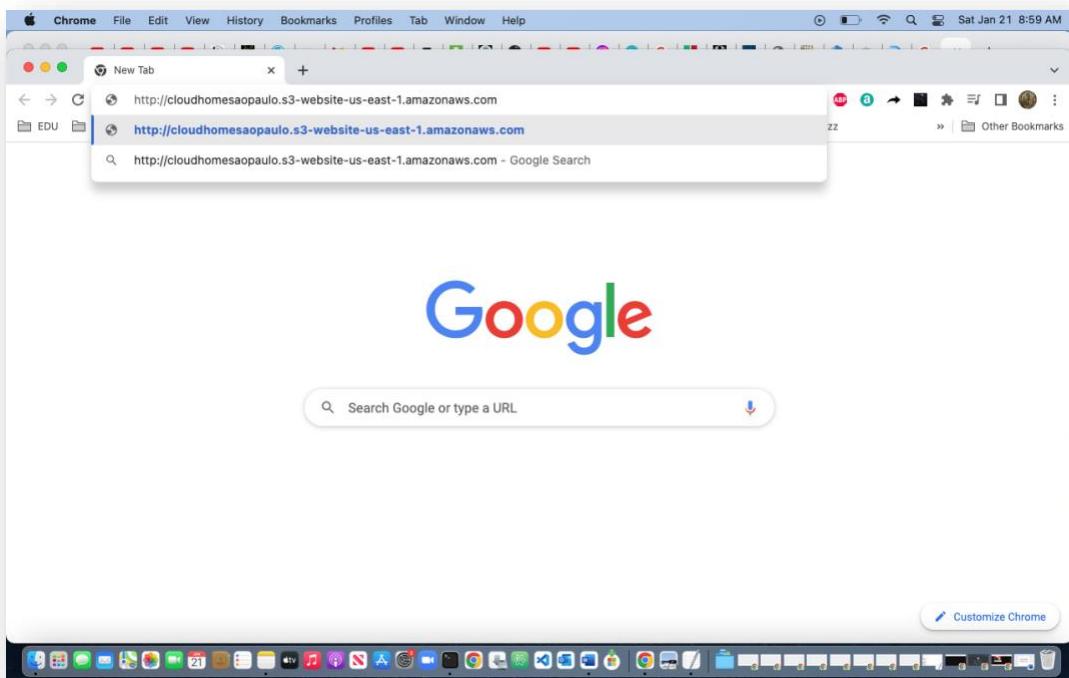
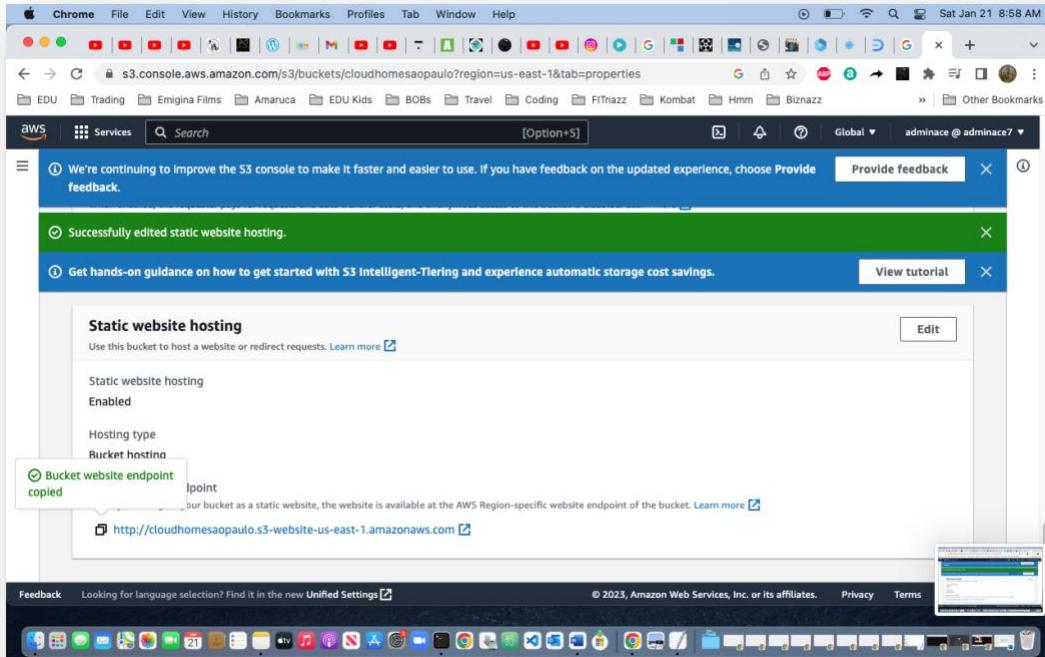


Below you can see the green success banner letting us know our website is up and running. Let us scroll, while under the **Properties** tab, back to the **Static Web Hosting** section.

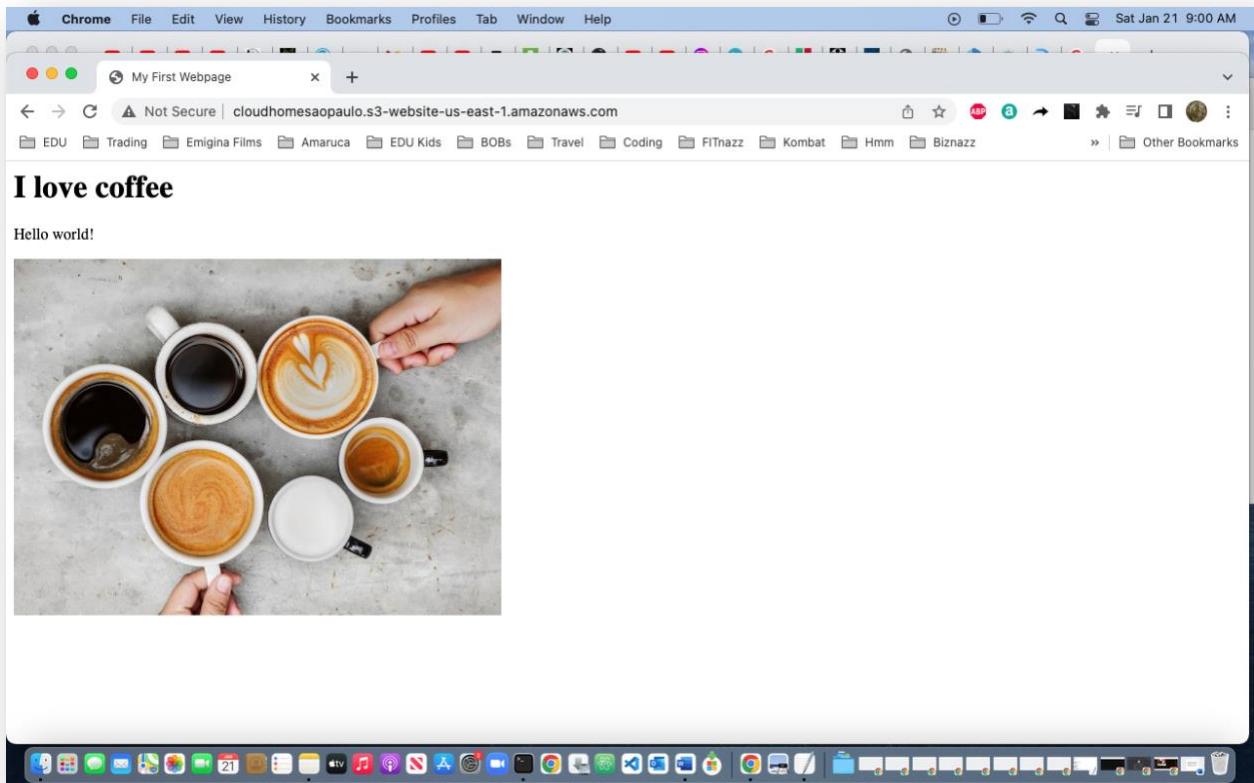
The screenshot shows the AWS S3 console in a web browser. The URL is `s3.console.aws.amazon.com/s3/buckets/cloudhomesaopaulo?region=us-east-1&tab=properties`. The top navigation bar includes links for EDU, Trading, Emigina Films, Amarica, EDU Kids, BOBs, Travel, Coding, FITnazz, Kombat, Hmm, and Biznazz. The main content area displays the 'Bucket overview' for the 'cloudhomesaopaulo' bucket. A green success banner at the top states 'Successfully edited static website hosting.' Below it, another message encourages users to 'Get hands-on guidance on how to get started with S3 Intelligent-Tiering and experience automatic storage cost savings.' The 'Properties' tab is selected in the navigation bar. The 'Static website hosting' section is visible, showing that it is currently enabled. The endpoint listed is `http://cloudhomesaopaulo.s3-website-us-east-1.amazonaws.com`.

This screenshot is identical to the one above, showing the AWS S3 console for the 'cloudhomesaopaulo' bucket. The 'Properties' tab is selected, and the 'Static website hosting' section is displayed. The 'Edit' button in the top right corner of this section is highlighted with a red box. The rest of the interface, including the success banner and the bucket details, remains the same.

Here we will click on the double boxes, copying our *Bucket website endpoint* address. From here we can open up a new web browser window or tab to paste the bucket website endpoint in the address bar.

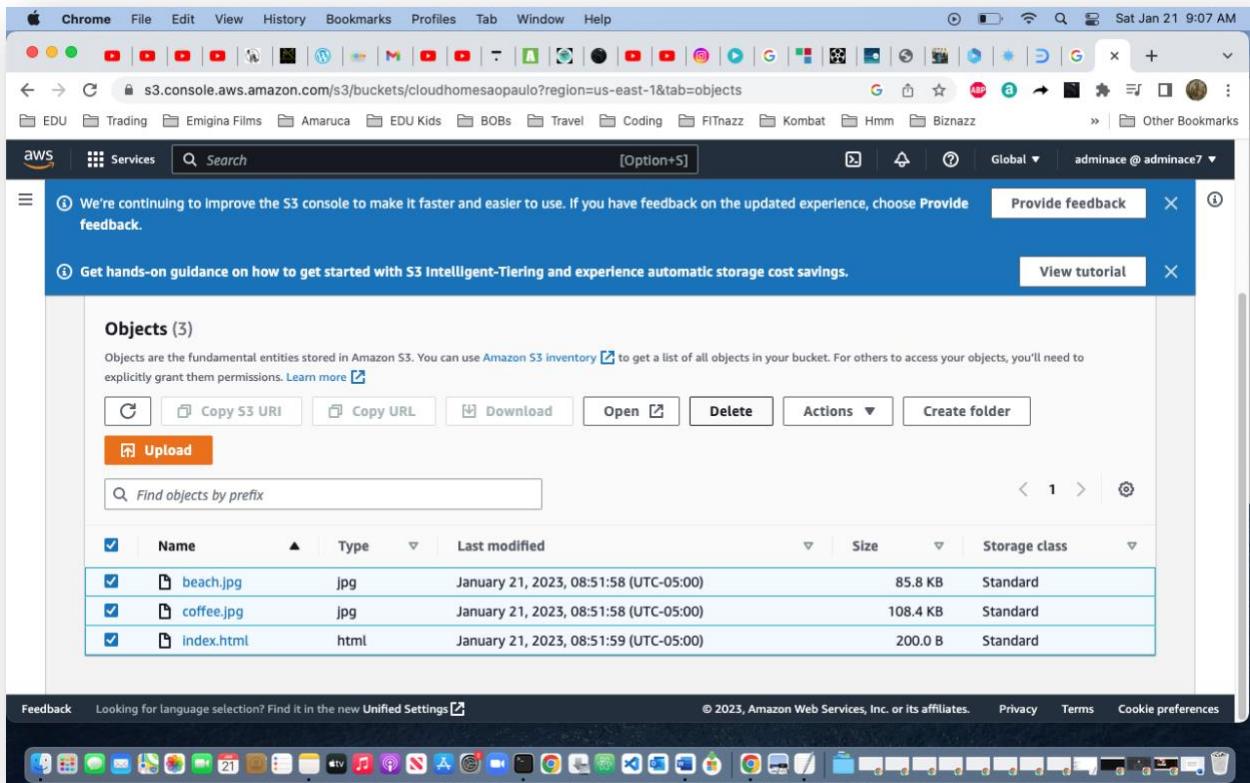


As we can see below, we are using the public internet to access our ACL, which allows traffic into our S3 bucket to see our objects on display on a web page.



How to Terminate and S3

To terminate our S3, we will delete the objects and then delete the bucket. Go to the **Objects** page and select all the objects checkboxes. Next click the “*delete*” button.



The screenshot shows a Chrome browser window displaying the AWS S3 console at s3.console.aws.amazon.com/s3/buckets/cloudhomesaopaulo?region=us-east-1&tab=objects. The browser's address bar and various tabs are visible at the top. The AWS logo and services navigation bar are at the top left. A blue banner at the top provides feedback and links to tutorials. The main content area is titled "Objects (3)". It lists three files: "beach.jpg", "coffee.jpg", and "index.html", each with a checkbox checked. Below the list are buttons for "Upload", "Find objects by prefix", and navigation controls. At the bottom, there are links for "Feedback", "Unified Settings", and copyright information, along with the Mac OS X dock at the very bottom.

| checkbox | Name | Type | Last modified | Size | Storage class |
|-------------------------------------|------------|------|--|----------|---------------|
| <input checked="" type="checkbox"/> | beach.jpg | jpg | January 21, 2023, 08:51:58 (UTC-05:00) | 85.8 KB | Standard |
| <input checked="" type="checkbox"/> | coffee.jpg | jpg | January 21, 2023, 08:51:58 (UTC-05:00) | 108.4 KB | Standard |
| <input checked="" type="checkbox"/> | index.html | html | January 21, 2023, 08:51:59 (UTC-05:00) | 200.0 B | Standard |

On the **Delete Objects** page, we will type “*permanently delete*” in the input field. Then select the “*delete objects*” button to confirm deletion. We will see a green success banner across the page on the next screen.

The screenshot shows the AWS S3 console on a Mac OS X desktop. The browser window title is "s3.console.aws.amazon.com/s3/buckets/clouhomesaopaulo/object/delete?region=us-east-1&showversions...". The main content area is titled "Delete objects". Under "Specified objects", there is a search bar with placeholder text "Find objects by name" and a pagination indicator "1". A warning box contains the following text:

- If a folder is selected for deletion, all objects in the folder will be deleted, and any new objects added while the delete action is in progress might also be deleted. If an object is selected for deletion, any new objects with the same name that are uploaded before the delete action is completed will also be deleted.
- Deleting the specified objects can't be undone.

At the bottom of the page, there are links for "Feedback", "Unified Settings", "Privacy", "Terms", and "Cookie preferences".

The screenshot shows the same AWS S3 console window. The main content area is titled "Delete objects". Below it, a section titled "Permanently delete objects?" contains a text input field with the placeholder "To confirm deletion, type *permanently delete* in the text input field." The input field contains the text "permanently delete". At the bottom of this section are two buttons: "Cancel" and a prominent orange "Delete objects" button.

Select the “*close*” button to exit this page. On the objects page, we will click the *buckets* hyperlink to go to our buckets page.

The screenshot shows a Chrome browser window with the AWS S3 console URL: <https://s3.console.aws.amazon.com/s3/buckets/cloudhomesaopaulo/object/delete?region=us-east-1&showversions...>. The page displays a success message: "Successfully deleted objects" with a count of "3 objects, 194.4 KB". A "Close" button is visible in the top right corner of the modal.

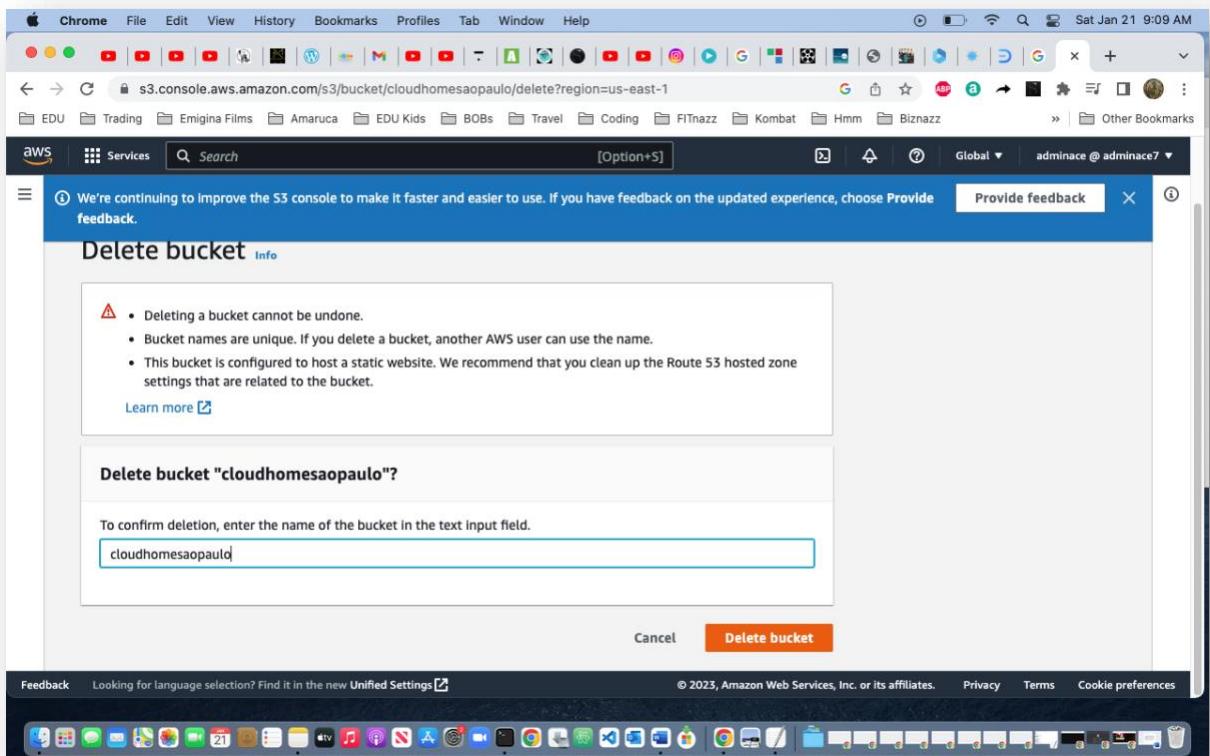
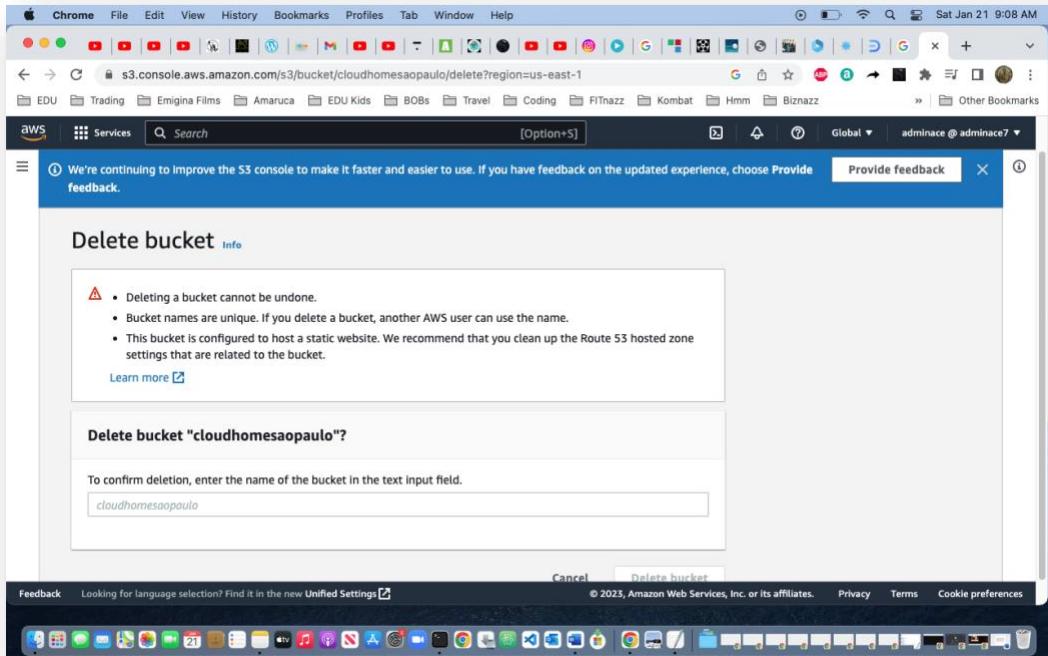
The screenshot shows a Chrome browser window with the AWS S3 console URL: <https://s3.console.aws.amazon.com/s3/buckets/cloudhomesaopaulo?region=us-east-1&tab=objects>. The page displays the "Objects" tab for the "cloudhomesaopaulo" bucket. It shows a summary table with three rows: "Source" (s3://cloudhomesaopaulo), "Successfully deleted" (3 objects, 194.4 KB), and "Failed to delete" (0 objects). Below the table, there is a "Feedback" section and a navigation bar at the bottom.

Here on the **Buckets** page, we will select the radio button next to the *bucket name*. Then we'll select the “*delete*” button.

The screenshot shows the AWS S3 Buckets page in a Chrome browser. The URL is s3.console.aws.amazon.com/s3/buckets?region=us-east-1®ion=us-east-1. The page includes a sidebar with options like Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Block Public Access settings for this account, Storage Lens, Dashboards, and AWS Organizations settings. The main content area features an Account snapshot section with a Storage Lens dashboard link. Below it is a Buckets (1) section with a table showing one bucket: clouhomesaopaulo (US East (N. Virginia) us-east-1). The table has columns for Name, AWS Region, Access, and Creation date. Buttons for Copy ARN, Empty, Delete, and Create bucket are available above the table. A search bar for Find buckets by name is also present. The bottom of the screen shows the Mac OS X dock with various application icons.

| Name | AWS Region | Access | Creation date |
|------------------|---------------------------------|-----------------------|--|
| clouhomesaopaulo | US East (N. Virginia) us-east-1 | Objects can be public | January 21, 2023, 08:49:38 (UTC-05:00) |

On the **Delete Bucket** page, type the bucket name in the input field provided below. Lastly, select the “*delete bucket*”, button.



We have successfully setup and torn down an S3 bucket with objects.

