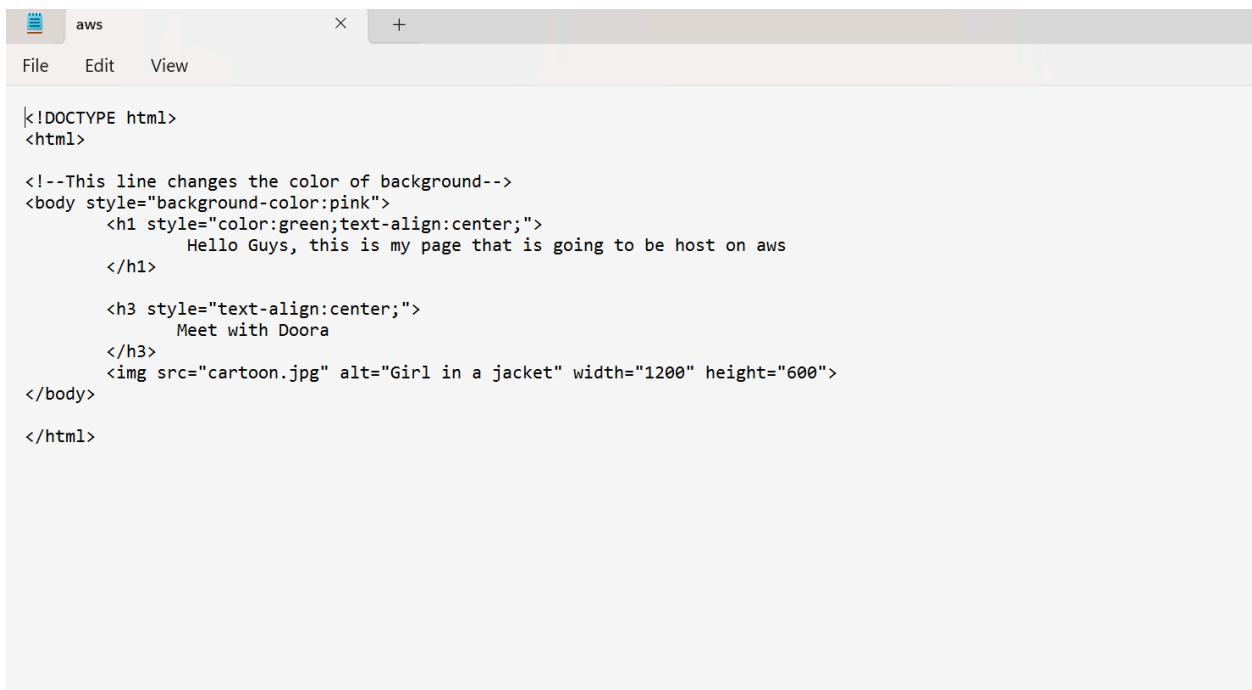


# Cloud AWS Task - Week 1

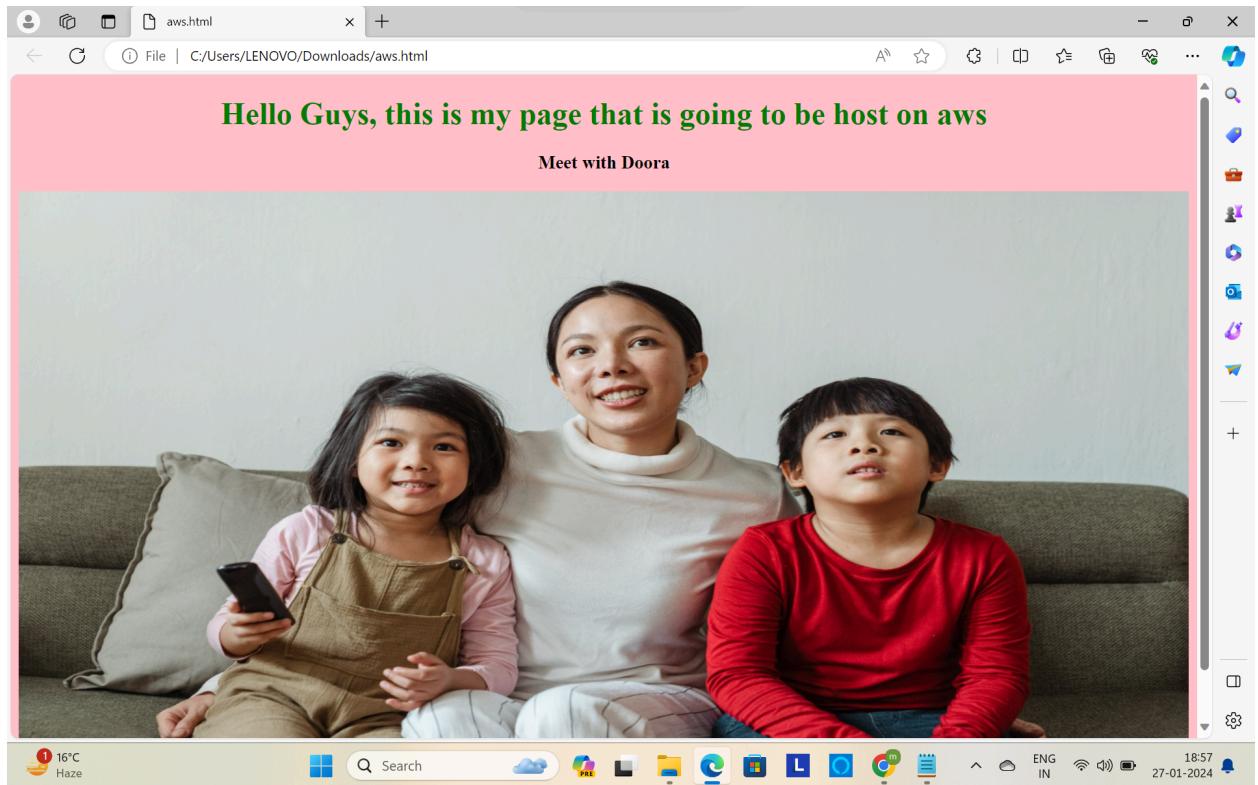
## Task : Set up a Simple Static Website on Amazon S3

### Step-1) Create a simple static website using HTML and CSS



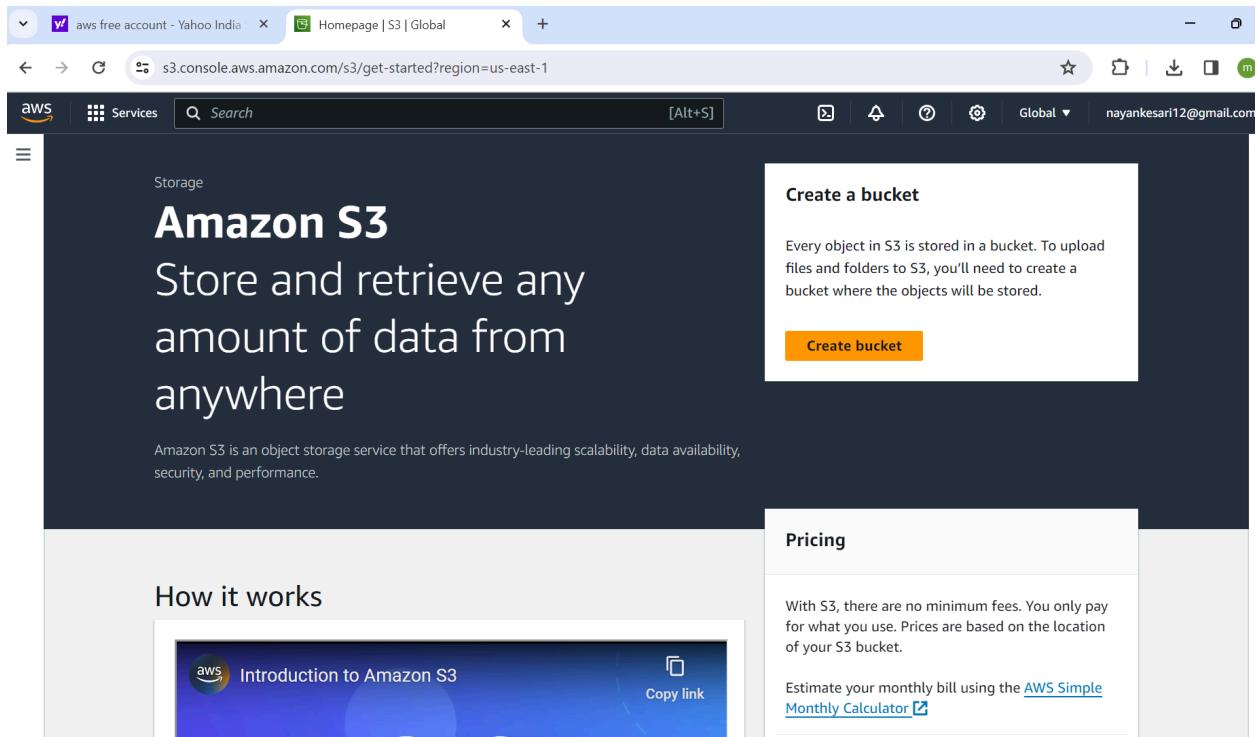
The screenshot shows a code editor window titled "aws". The menu bar includes "File", "Edit", and "View". The main pane displays the following HTML code:

```
<!DOCTYPE html>
<html>
    <!--This line changes the color of background-->
    <body style="background-color:pink">
        <h1 style="color:green;text-align:center;">
            Hello Guys, this is my page that is going to be host on aws
        </h1>
        <h3 style="text-align:center;">
            Meet with Doora
        </h3>
        
    </body>
</html>
```



Now Go to aws free account and create an account

## Step-2) Search S3 instance then click on it and then create a bucket



Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. Customers of all sizes and industries can use Amazon S3 to store and protect any amount of data for a range of use cases, such as data lakes, websites, mobile applications, backup and restore, archive, enterprise applications, IoT devices, and big data analytics.

Enter the bucket name

The screenshot shows the AWS S3 'Create bucket' interface. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, a search bar, and various icons. Below it, the breadcrumb trail shows 'Amazon S3 > Buckets > Create bucket'. The main title is 'Create bucket' with an 'Info' link. A sub-instruction says 'Buckets are containers for data stored in S3.' with a 'Learn more' link. The 'General configuration' section is open, showing the 'AWS Region' set to 'US East (N. Virginia) us-east-1'. Under 'Bucket type', 'General purpose' is selected (indicated by a blue border), while 'Directory - New' is also listed with its description. The 'Bucket name' field contains 'mystatic-website-on-aws-s3'. A note below it states: 'Bucket name must be unique within the global namespace and follow the bucket naming rules.' with a 'See rules for bucket naming' link. There's also a link 'Copy settings from existing bucket - optional'.

Uncheck Block Public Access setting, because by default Amazon S3 blocks public access to your account and buckets.

The screenshot shows the 'Block Public Access settings for this bucket' section. It includes a note about public access being granted through ACLs, policies, and point policies. A link to 'Learn more' is provided. Below are four checkboxes for different access types:

- 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.

A yellow warning box at the bottom states: '⚠️ Turning off block all public access might result in this bucket and the objects within becoming public'. It also says 'AWS recommends that you turn on block all public access, unless public access is required for specific and'.

Now keep everything default and click on create bucket.

The screenshot shows the 'Default encryption' section. It notes that server-side encryption is automatically applied to new objects stored in this bucket. The 'Encryption type' section shows 'Server-side encryption with Amazon S3 managed keys (SSE-S3)' selected. The 'Bucket Key' section shows 'Enable' selected. A 'Advanced settings' section is partially visible. At the bottom, a note says 'After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.' The 'Create bucket' button is highlighted in orange.

Now we can see that the bucket has created successfully.

The screenshot shows the AWS S3 Buckets page. At the top, a green banner indicates that a bucket has been successfully created. Below the banner, the 'Account snapshot' section is visible, followed by tabs for 'General purpose buckets' and 'Directory buckets'. Under the 'General purpose buckets' tab, there is a table listing one bucket:

Name	AWS Region	Access	Creation date
mystatic-website-on-aws-s3	US East (N. Virginia) us-east-1	Objects can be public	January 27, 2024, 19:05:08 (UTC+05:30)

Now go to your bucket that you have created

The screenshot shows the AWS S3 Bucket details page for 'mystatic-website-on-aws-s3'. The top navigation bar shows the bucket name. The main content area is titled 'mystatic-website-on-aws-s3' and includes tabs for 'Objects', 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. The 'Objects' tab is selected, showing a table with one row:

Name	Type	Last modified	Size	Storage class
No objects				

Below the table, a message states 'No objects' and 'You don't have any objects in this bucket.' A large 'Upload' button is prominently displayed.

Now Upload your static website here.

The screenshot shows the AWS S3 'Upload' interface. At the top, there's a search bar and a 'Global' dropdown set to 'nayankesari12@gmail.com'. Below the header, the word 'Upload' is displayed with an 'Info' link. A note below the header says: 'Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)'.

A large dashed box area is labeled 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.'

Below this, a table titled 'Files and folders (2 Total, 2.0 MB)' lists two items:

	Name	Folder	Type
<input type="checkbox"/>	aws.html	-	text/html
<input type="checkbox"/>	cartoon.jpg	-	image/jpeg

Buttons for 'Remove', 'Add files', and 'Add folder' are located above the table. A search bar labeled 'Find by name' is present. Below the table, a section titled 'Destination' shows the URL: `<3://mystatic-website-on-aws-s3`.

Website has uploaded successfully

The screenshot shows the 'Objects' tab of the 'mystatic-website-on-aws-s3' bucket in the AWS S3 console. The top navigation bar includes 'Amazon S3 > Buckets > mystatic-website-on-aws-s3' and tabs for 'Objects', 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'.

The 'Objects (2) Info' section shows two objects:

	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	aws.html	html	January 27, 2024, 19:08:23 (UTC+05:30)	420.0 B	Standard
<input type="checkbox"/>	cartoon.jpg	jpg	January 27, 2024, 19:08:27 (UTC+05:30)	2.0 MB	Standard

Actions available for each object include Copy S3 URI, Copy URL, Download, Open, Delete, Actions (with a dropdown), Create folder, and Upload. A note at the bottom states: 'Objects are the 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](#)'.

**Step-3)** In the **Bucket list**, choose the **name of the bucket** that you want to **enable** static website **hosting** for. Choose **Properties** and then Under **Static Website Hosting**, choose **Edit**.

The screenshot shows the AWS S3 Bucket Properties page for a bucket named "mystatic-website-on-aws-s3". The "Static website hosting" section is expanded, showing the "Edit" button. The "Hosting type" dropdown is set to "Host a static website". A callout box highlights the "For your customers to access content at the website endpoint, you must make all your content publicly readable." note.

**Object Lock**  
Store objects using a write-once-read-many (WORM) model to help you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely. Object Lock works only in versioned buckets. [Learn more](#)

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

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

Under **Static website hosting**, choose **Enable**

The screenshot shows the "Edit static website hosting" configuration page. The "Hosting type" dropdown is set to "Host a static website". A callout box highlights the note about making content publicly readable.

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

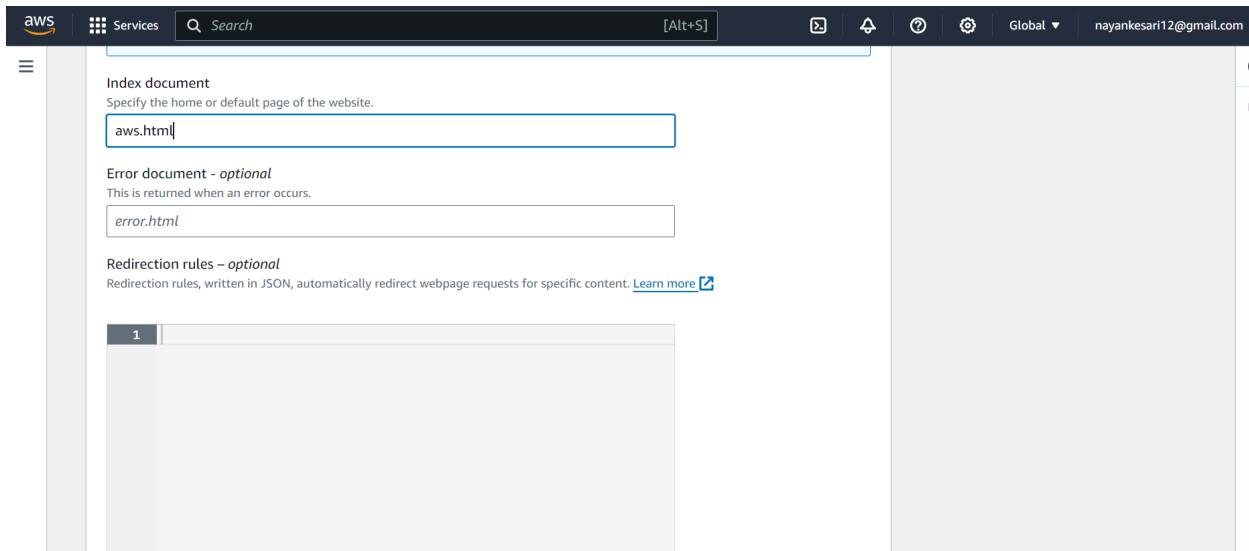
**Hosting type**

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

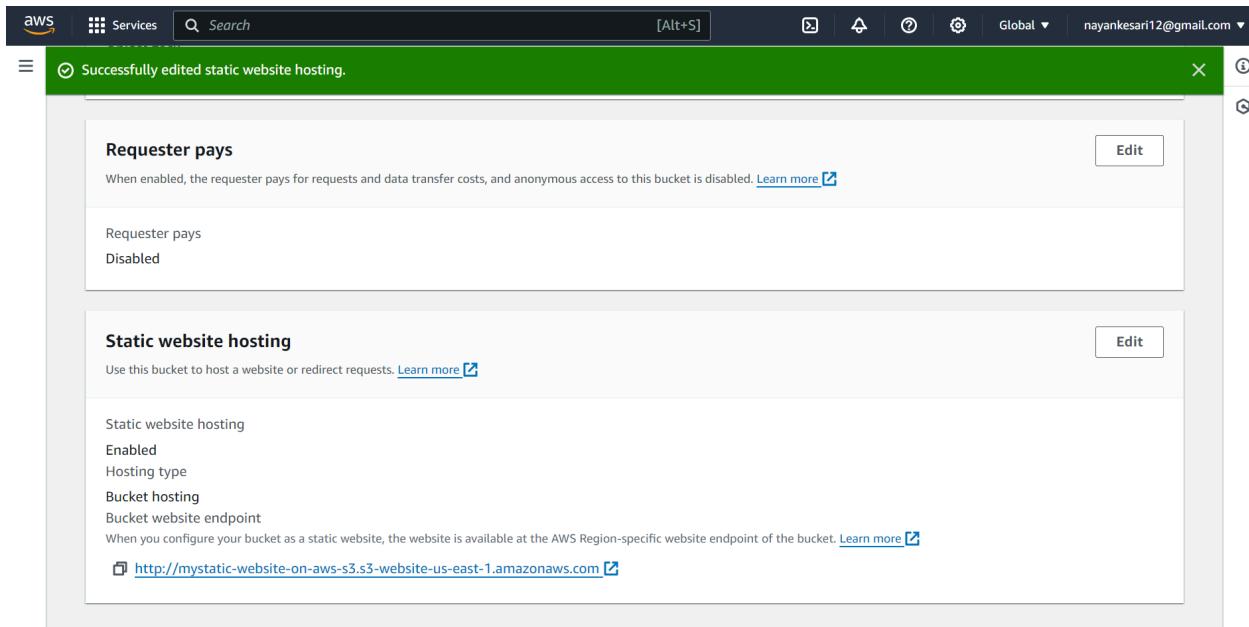
Redirect requests for an object  
Redirect requests to another bucket or domain. [Learn more](#)

**Index document**

In **Index document**, enter the file name of the index document and then click on **Save changes**



Now Check whether your website is accessible or not . So copy the url given and paste on the browser and check.



We can see that it's not accessible

Not secure mystatic-website-on-aws-s3.s3-website-us-east-1.amazonaws.com

## 403 Forbidden

- Code: AccessDenied
- Message: Access Denied
- RequestId: YGAGDE7D3RPQWT4T
- HostId: IKIZtyXllkk7C674KgIYFstBwrW/m14Si/yu5bLAQAE44lZViL/wkwAXcjo+w+BvMGyRgJSwQdw=

So here we can say that the bucket is publicly accessible but the objects that are inside the bucket are not publicly accessible. So for that follow step 4.

#### Step 4) Add a bucket policy that makes your bucket content publicly available

Go to your **bucket** then go to **permissions** then under **Bucket Policy** choose **Edit**

Amazon S3 > Buckets > mystatic-website-on-aws-s3

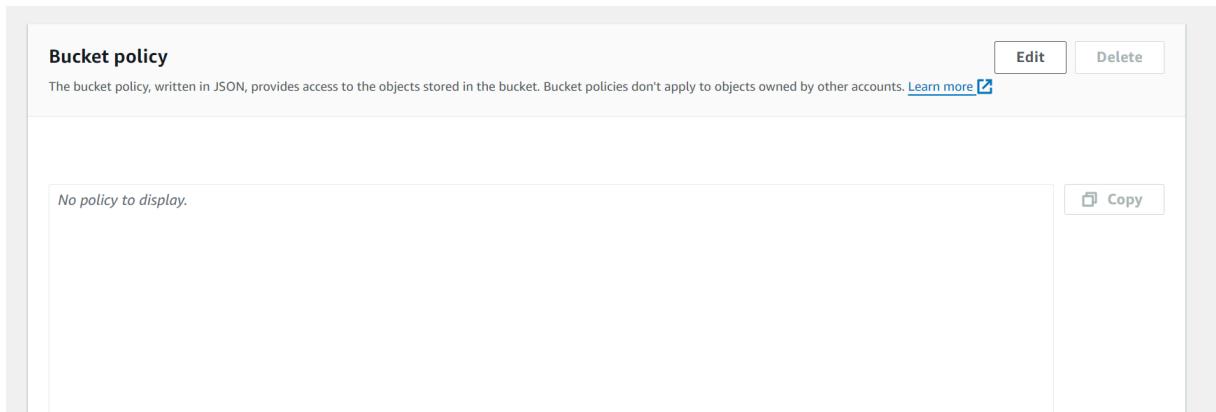
### mystatic-website-on-aws-s3 [Info](#)

Objects (2) [Info](#)

[Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Objects are the 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](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input checked="" type="checkbox"/>	aws.html	html	January 27, 2024, 19:08:23 (UTC+05:30)	420.0 B	Standard
<input type="checkbox"/>	cartoon.jpg	jpg	January 27, 2024, 19:08:27 (UTC+05:30)	2.0 MB	Standard



To grant public read access for your website, write this in the **bucket policy editor**. And then click on **Save changes**

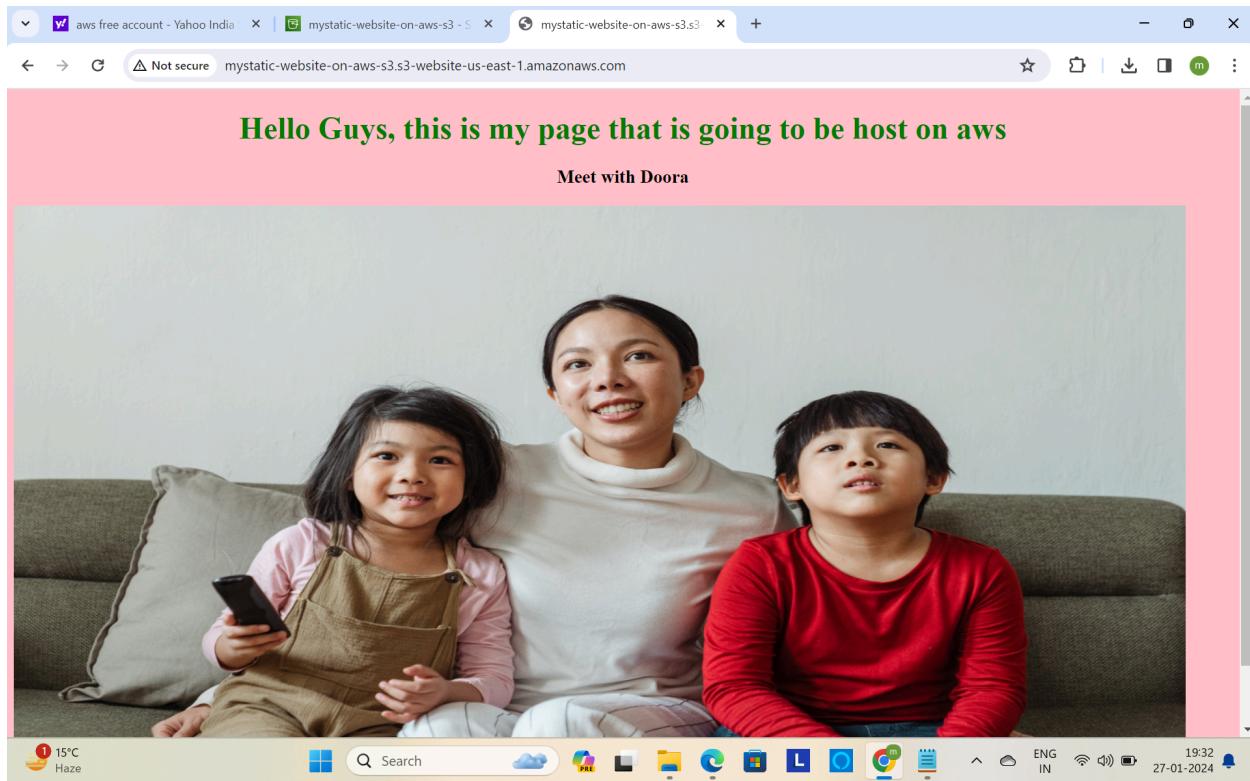
The screenshot shows the AWS Bucket policy editor interface. The policy JSON is displayed in a code editor-like area:

```
1▼ {
2  "Id": "Policy1706363288615",
3  "Version": "2012-10-17",
4  "Statement": [
5    {
6      "Sid": "Stmt1706363249326",
7      "Action": [
8        "s3:GetObject"
9      ],
10     "Effect": "Allow",
11     "Resource": "arn:aws:s3:::mystatic-website-on-aws-s3/*",
12   },
13   "Principal": [
14     "AWS": [
15       "arn:aws:s3:::*"
16     ]
17   }
18 ]
19 }]
```

Line 14 is highlighted with a red circle and the number "14". To the right of the code editor, there are buttons for "Edit statement", "Select a statement", and "+ Add new statement".

## Step - 5) Test your Website

Go to your bucket and then copy the url that is given to open your static website.



We can see that our website has been hosted successfully.

### Step-6) Clean up

Delete the AWS resources that you allocated so that you no longer accrue charges.

