***Static bulletproof Website Hosting Using AWS S3 Service***

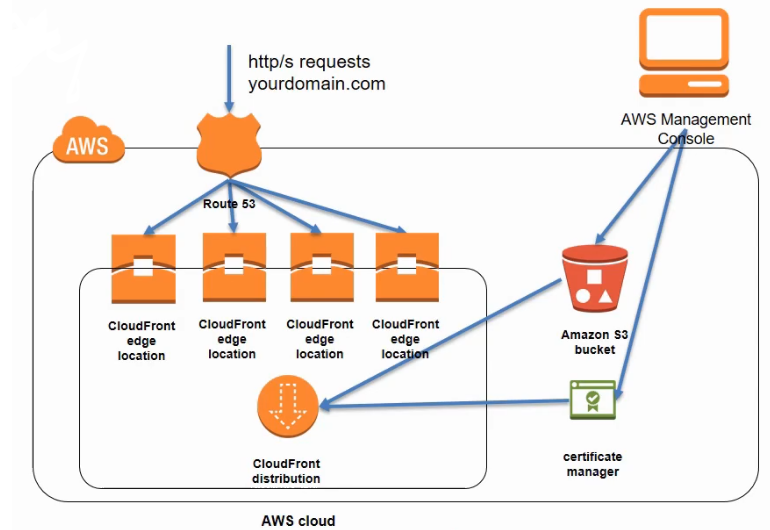
**What is Amazon S3?**

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

🡪 Store and protect any amount of data for a range of use cases, such as data lakes, websites, cloud-native applications, backups, archive, machine learning, and analytics.

🡪 Amazon S3 is designed for 99.999999999% (11 9's) of durability, and stores data for millions of customers all around the world.

**Architecture of this project**

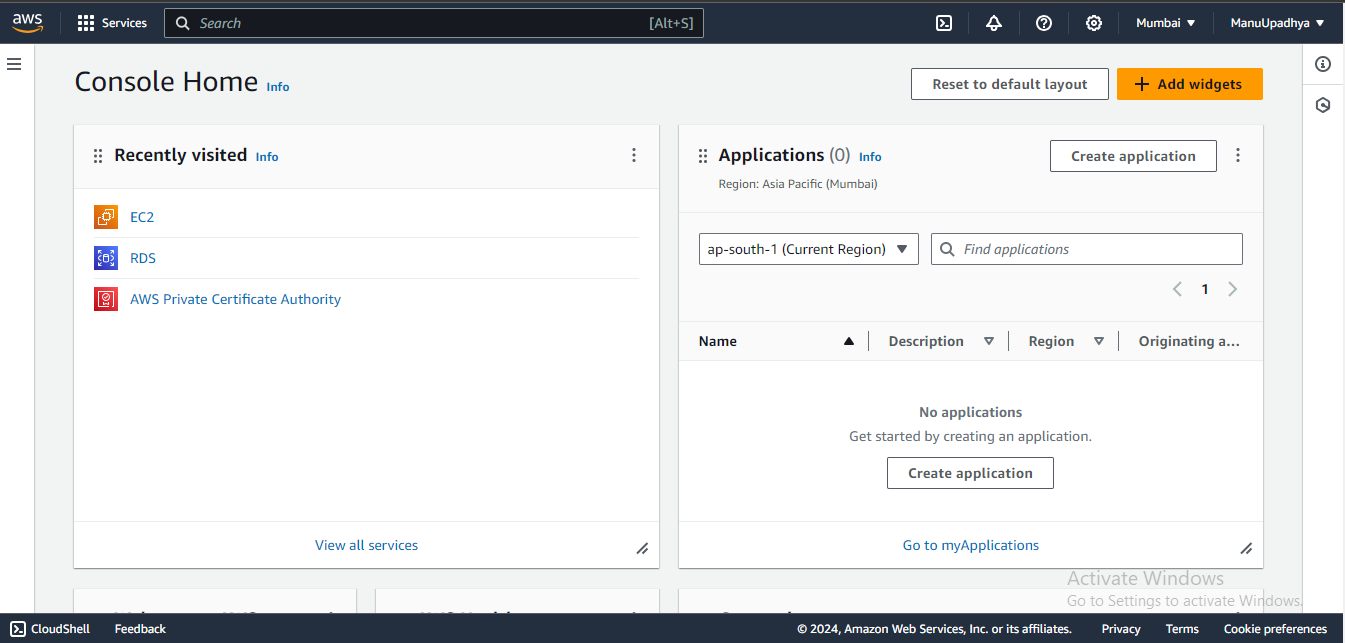
****

* Network Requests or Network Traffic (HTTP requests to access Our Static Website) to our Domain Name will come to Amazon Web Services(AWS) through **AWS Route 53** Service.
* **AWS Route 53** distributes requests to **AWS CloudFront edge location** close to our end user.
* **CloudFront distribution** will have a copy our Static Website. **CloudFront distribution** will copy the website from **AWS S3 bucket**, where the original static website is hosted.
* **CloudFront distribution** Updates the copy (of hosted Static website) it contains in regular time interval (example once in a day) from **AWS S3 bucket**. and distributes the updated copy across **AWS CloudFront edge location**.
* We will also be going to have HTTPS enabled, so we’re going to have SSL encryption on traffic coming to and from our website. For this we’re going to use create SSL certificate using **AWS certificate Manager (ACM)** service.

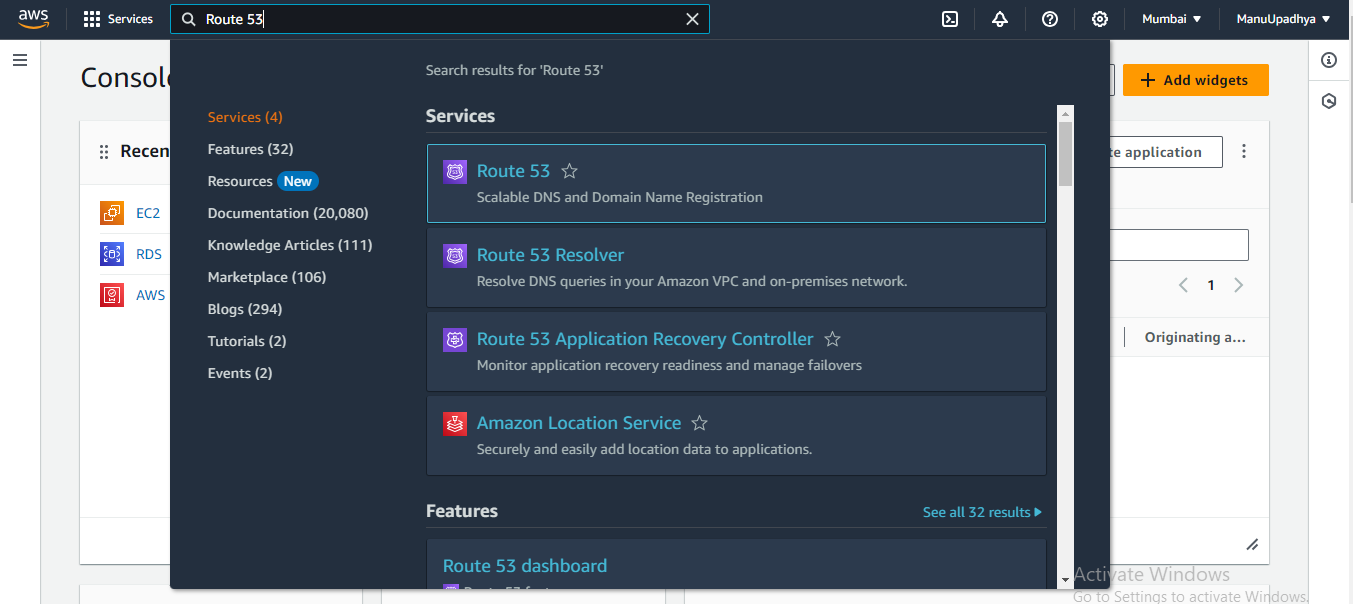
**Steps of Hosting the static Bullet Proof Website**

1. **Purchasing Domain Names With AWS Route 53**

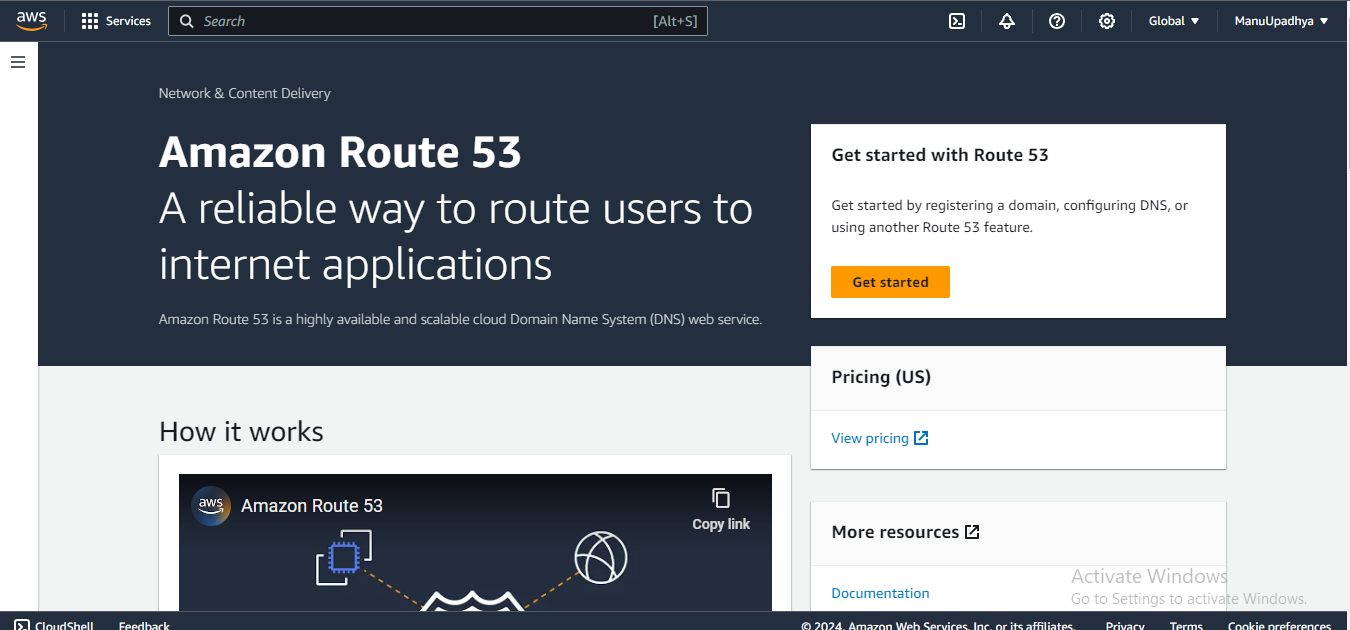
Log in to Your AWS Management Console.



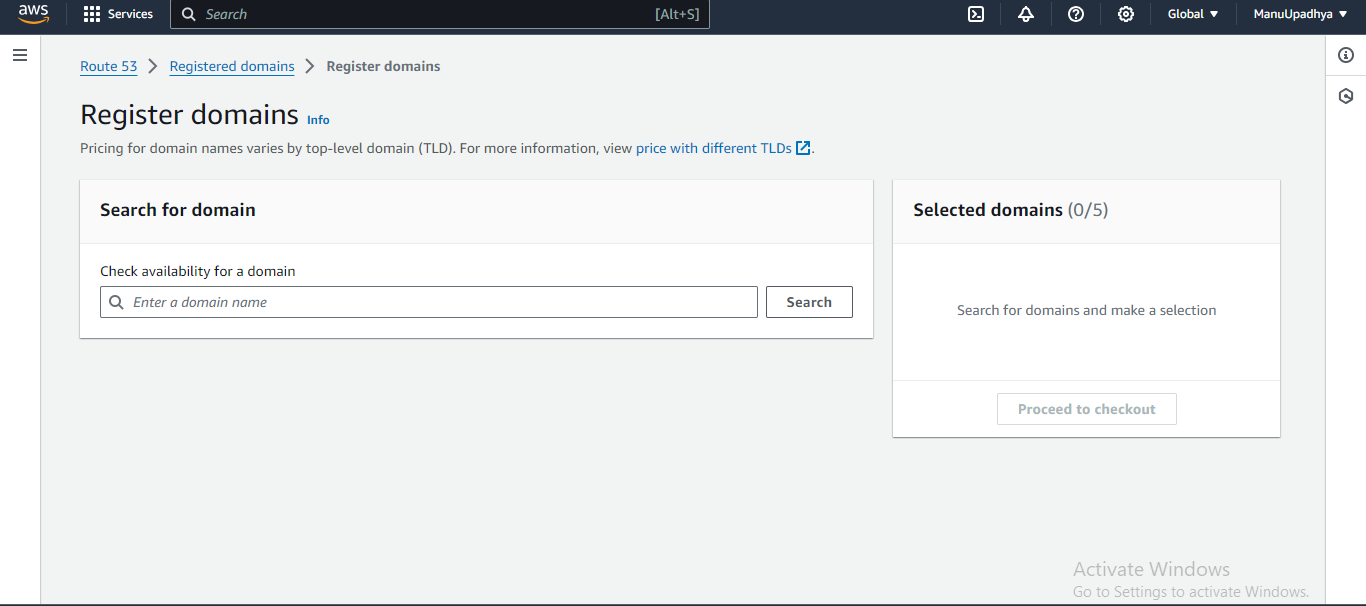
Navigate to Services then Search for **Route 53**.



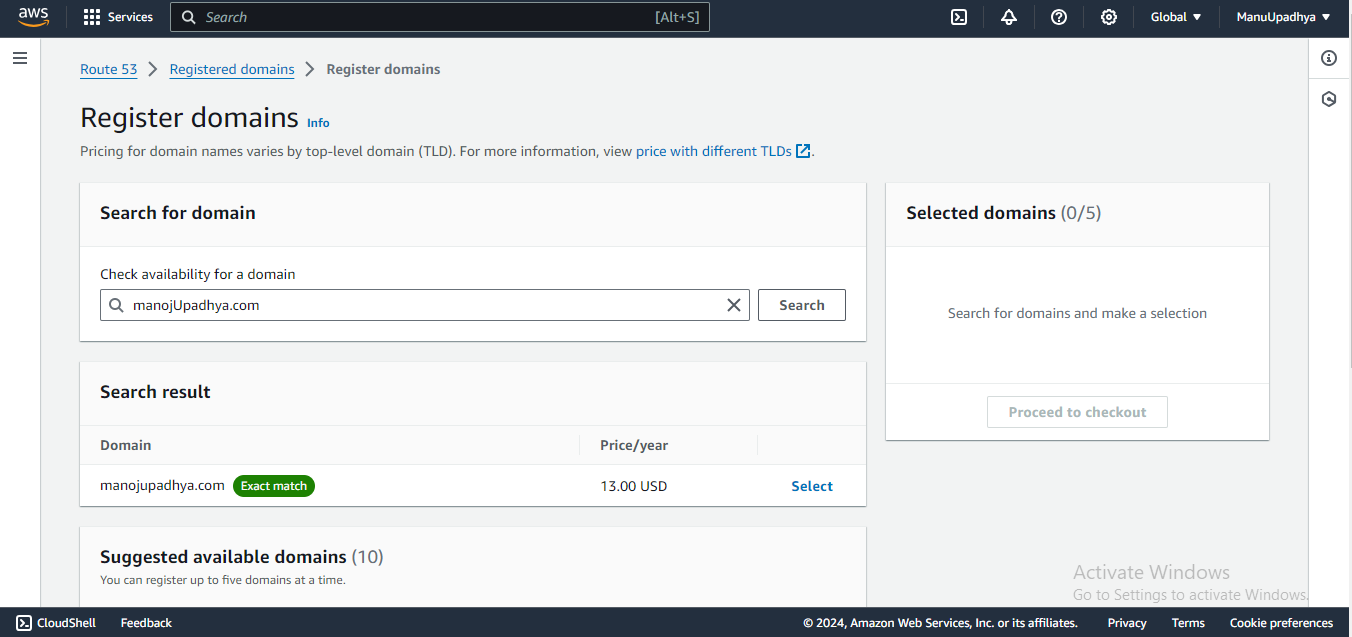
Now Click on **Route 53** to navigate to **Route 53 management console.**



Click on “**Get Started**” Then click on “**Register Domain**”.



Now Search for the domain name which we want to give to our website, consider for example “manojupadhya.com”



If we get the exact match, it is good or we must make some tweaks so that we will have domain name available, procced to checkout and buy the domain name.

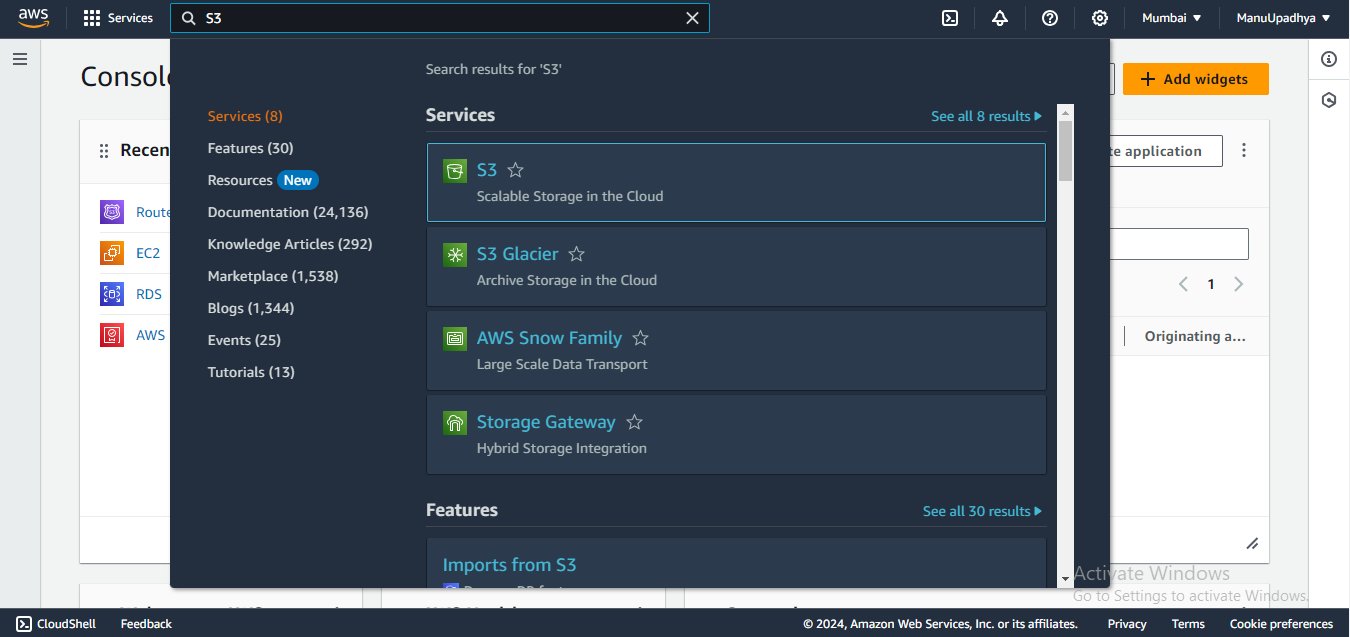


After successful purchase of domain name, the purchased domain name is visible under “Registered Domain names” tab.

[ I am not going to buy domain name here].

1. **Creating an S3 Bucket and Hosting our Website**

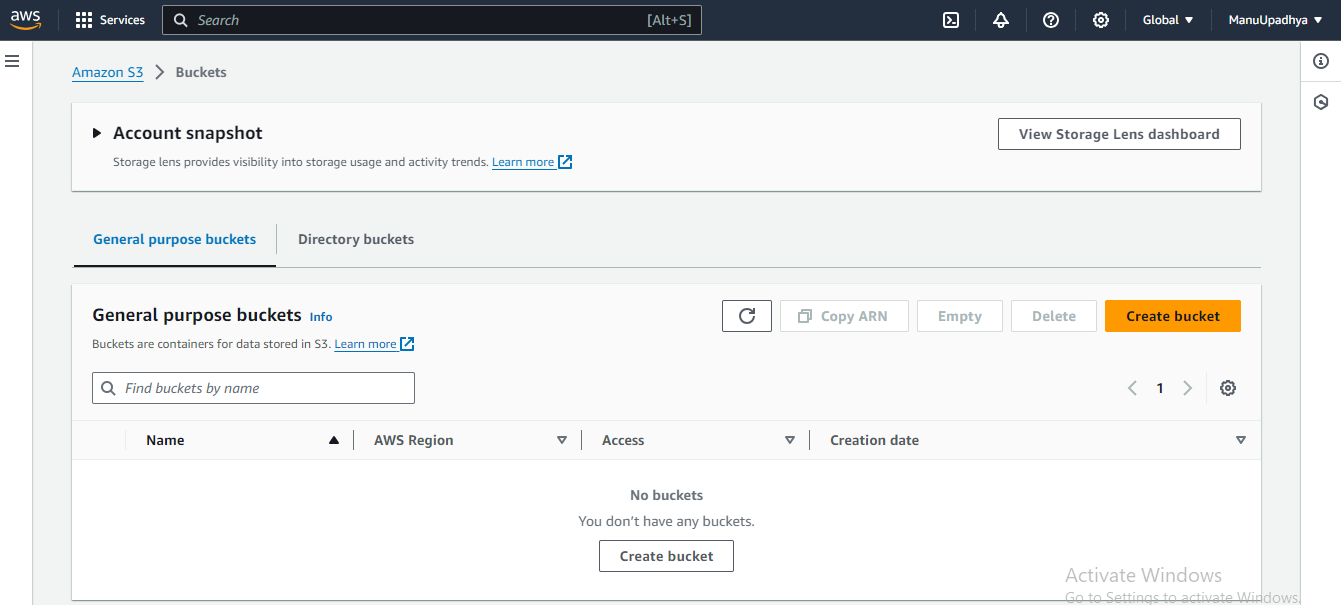
Navigate to “Services” then to “**AWS S3 Service**”.



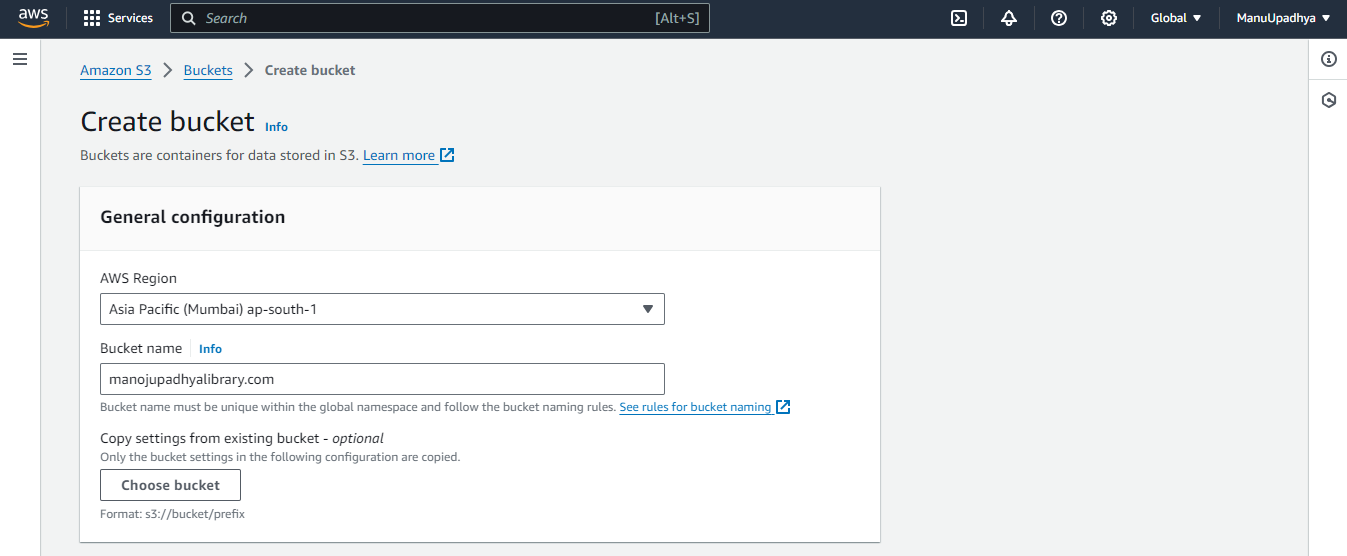
Click on “S3” to navigate to S3 management console.



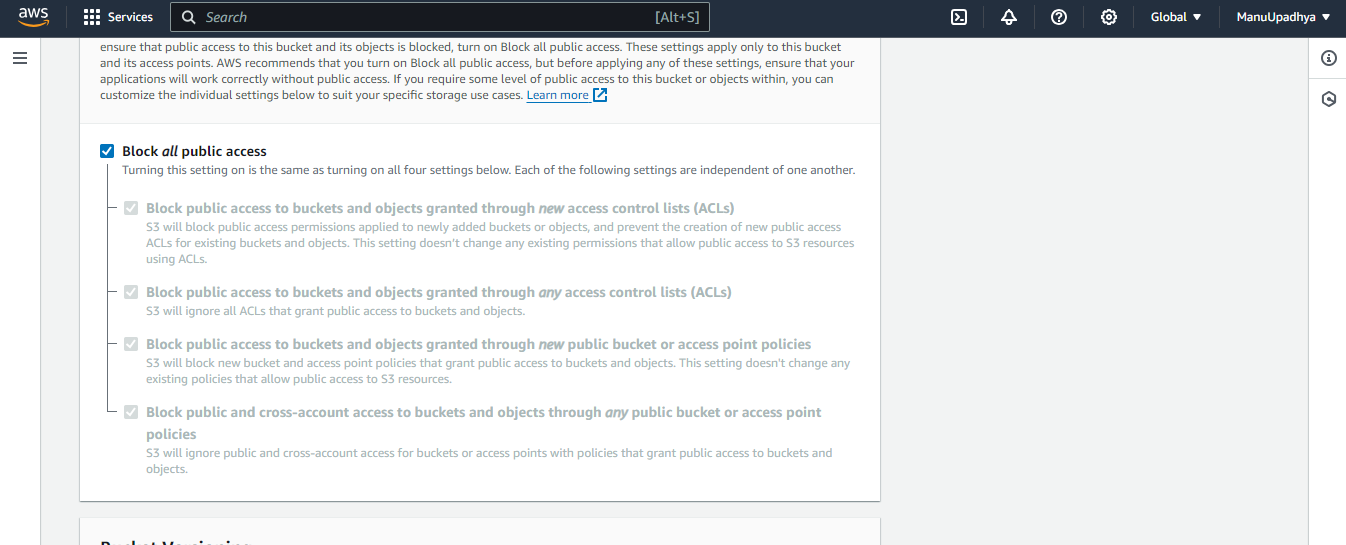
Click on “buckets” to view all previously created buckets.

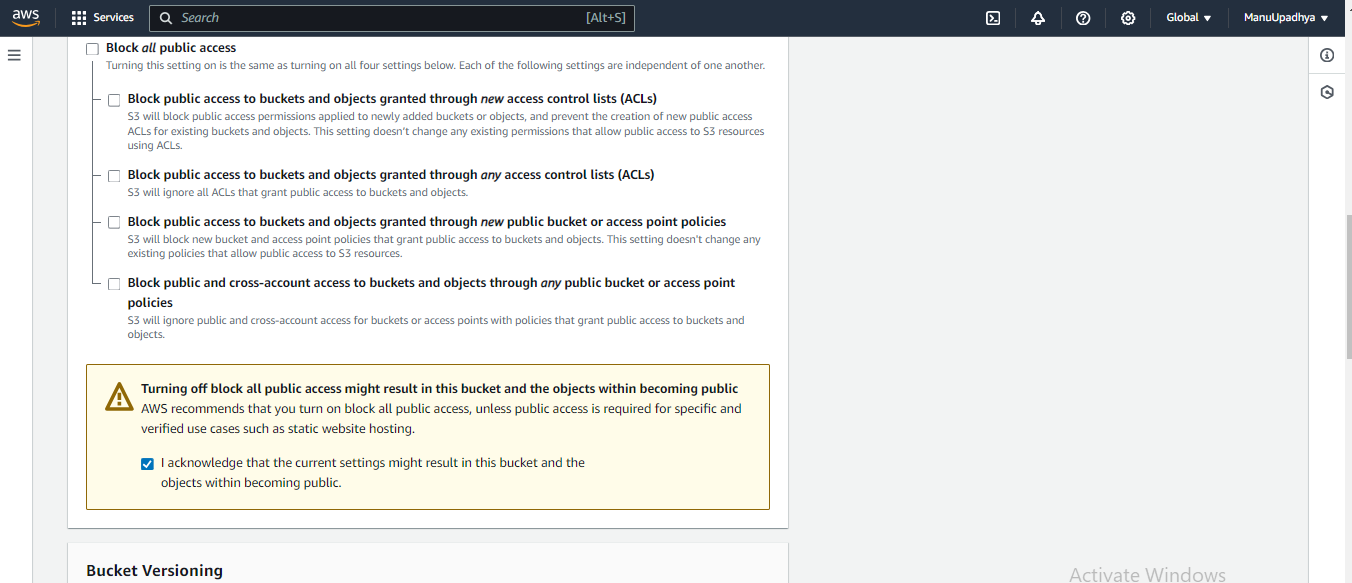


Now we need to create a new bucket to host our Website. Now Click on “Create Bucket”, Give our DNS website name (that we purchased) as bucket name. Select a Region ( doesn’t matter as S3 is a global Service)

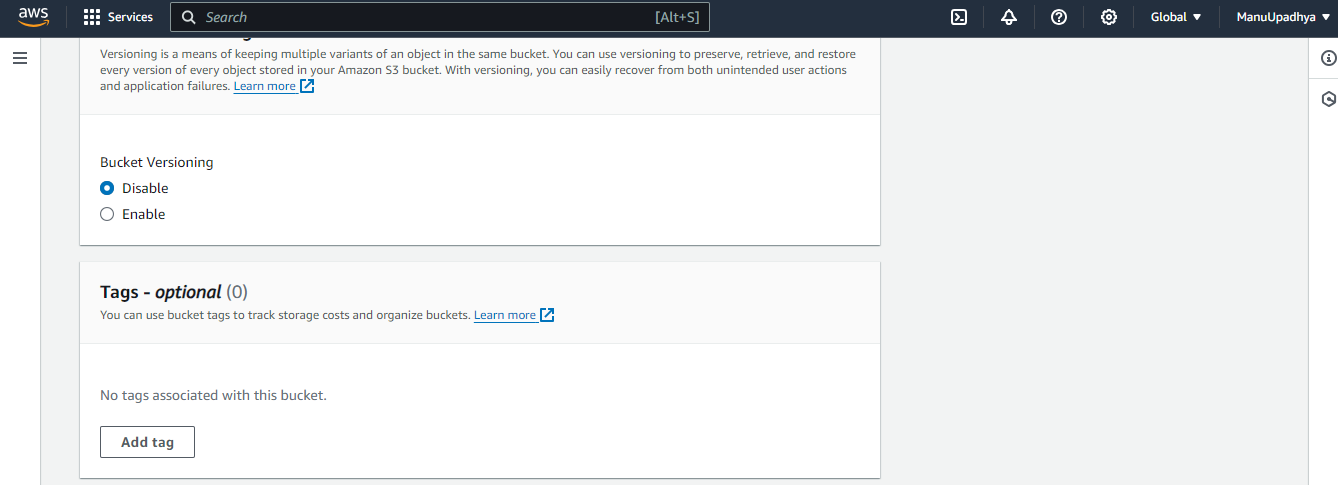


Disable the “Block all Public access” Option so that public should be able to access our websites.



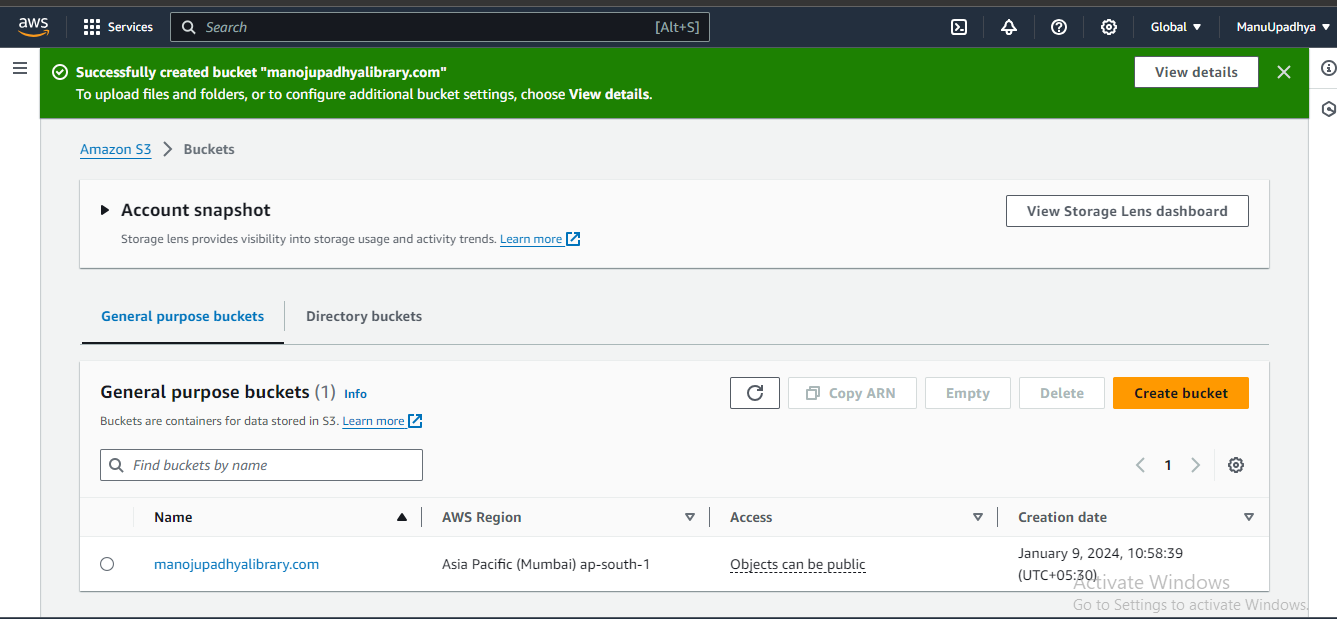


Make **Bucket Versioning** as Disabled, it will be enabled by default and keep **tags** as it is.

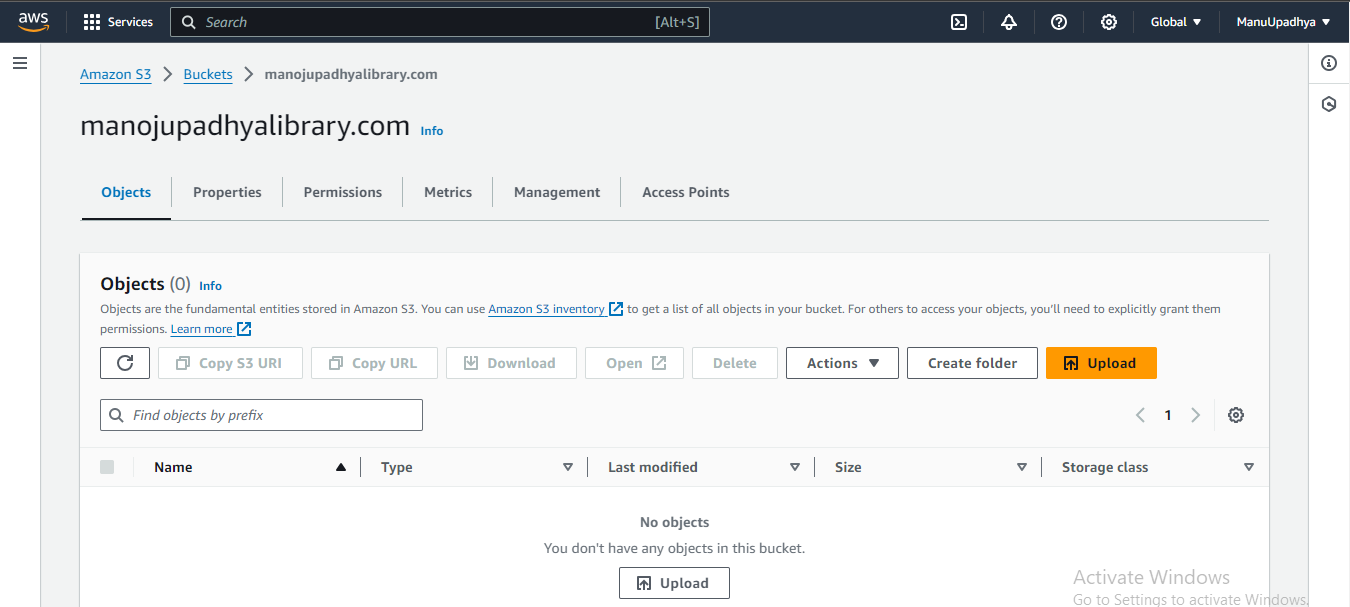


Now Click on “Create Bucket”.

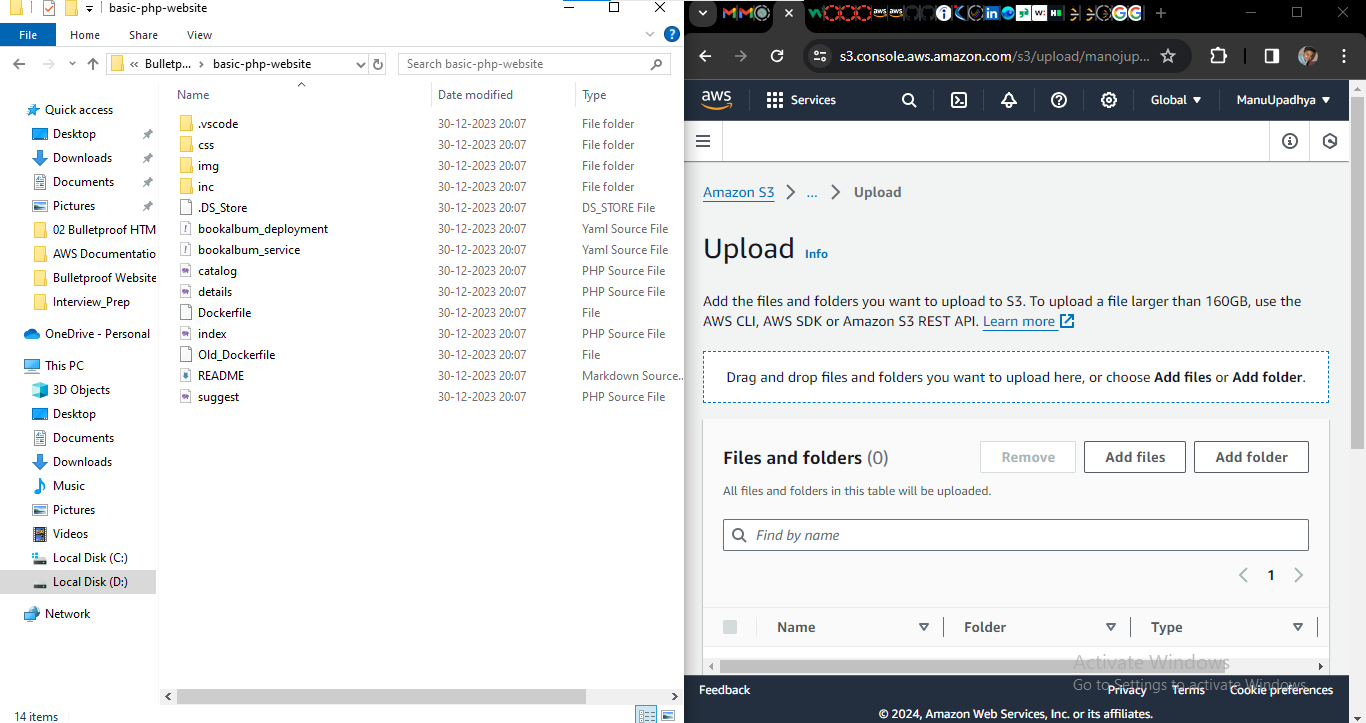
A new S3 bucket creation completed successfully. It will automatically navigate to “buckets” page where we can see our newly created bucket.



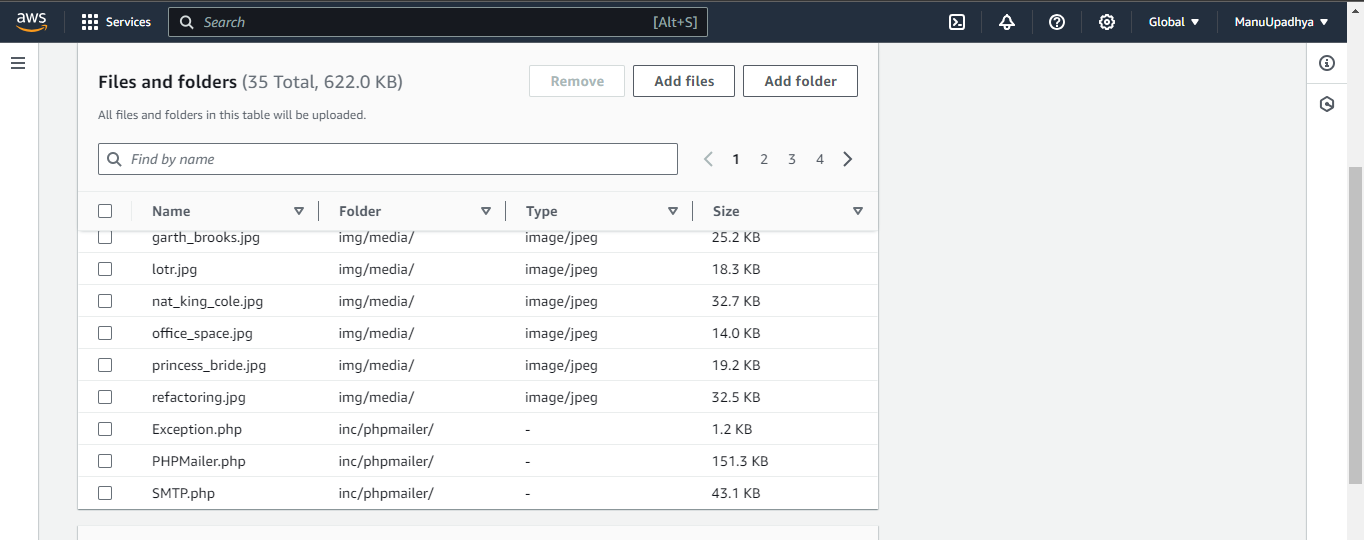
Now open the created bucket to upload our website.



Click on Upload. Drag and drop the entire website folder to S3 bucket.

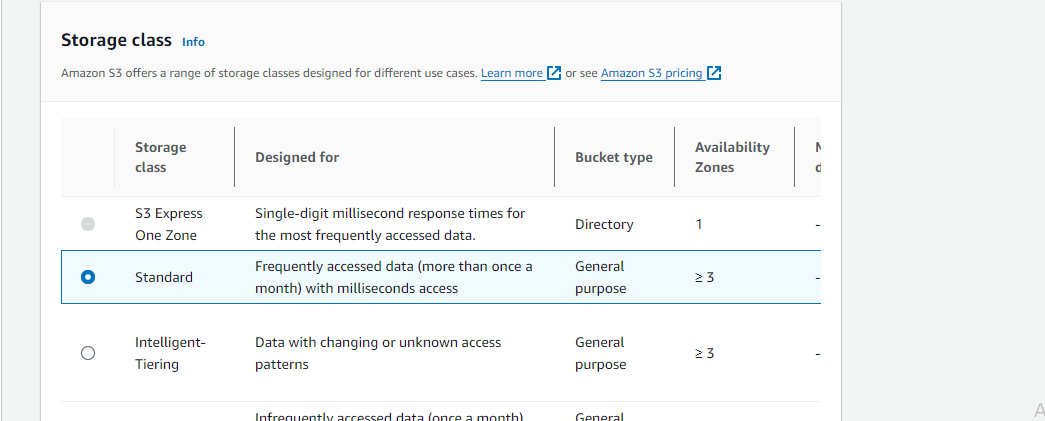


Uploaded website data in S3 bucket

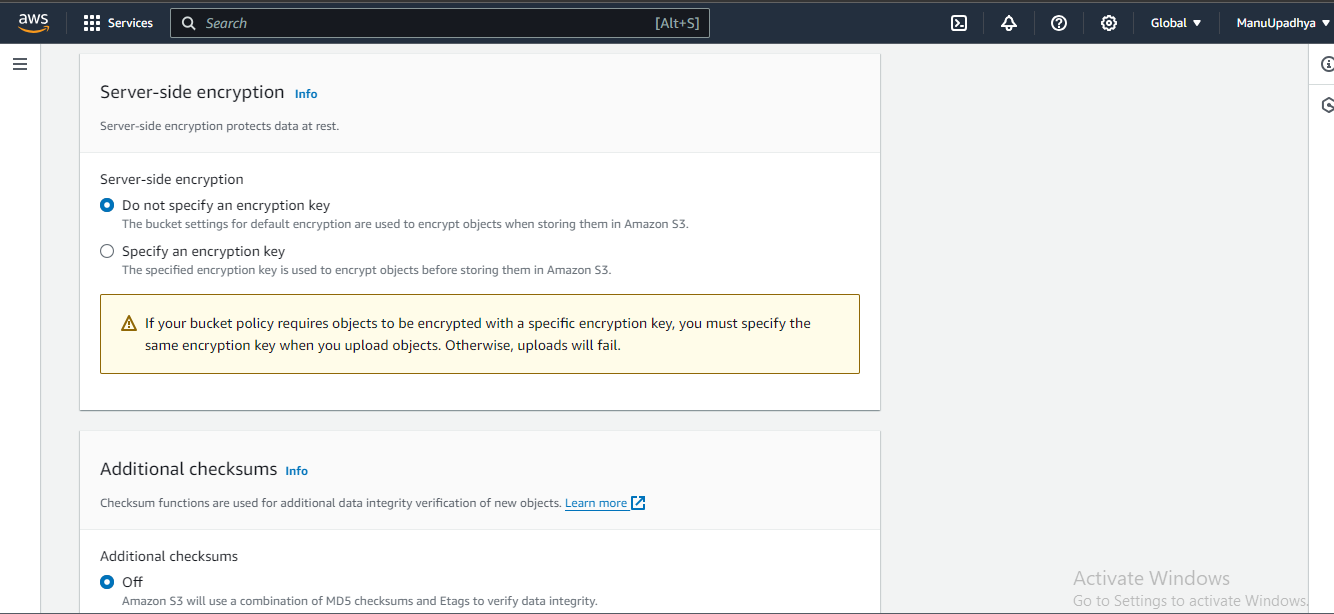


**Permissions:-** as we set bucket to have public access here, we don’t need to giver permissions.

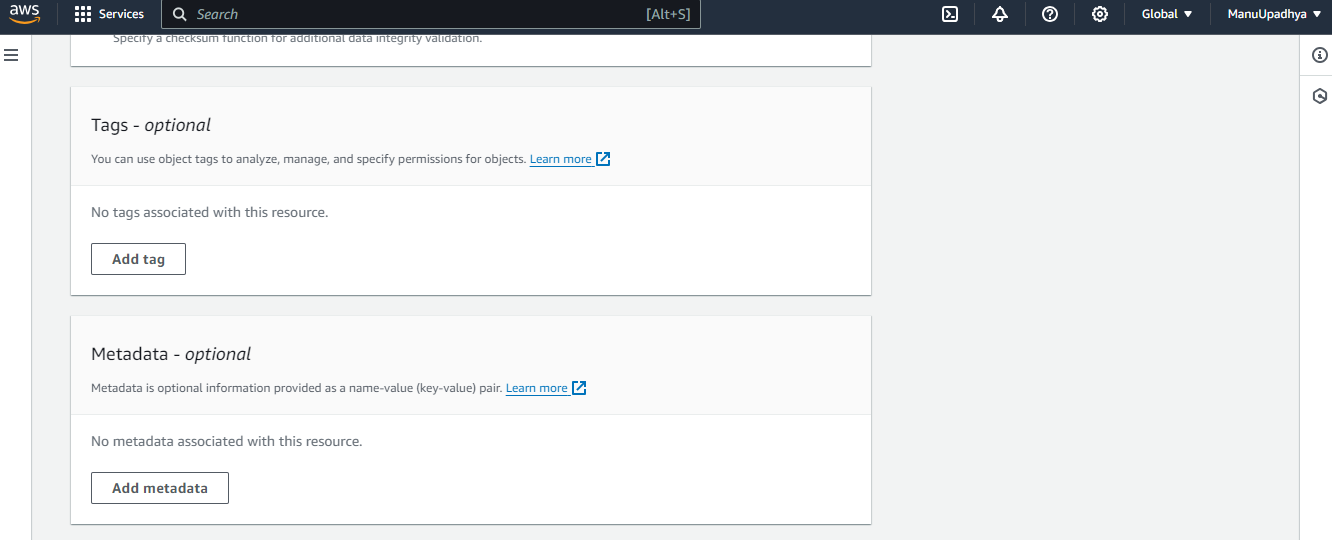
**Properties:-** Keep it default that is **“Standard”**

****

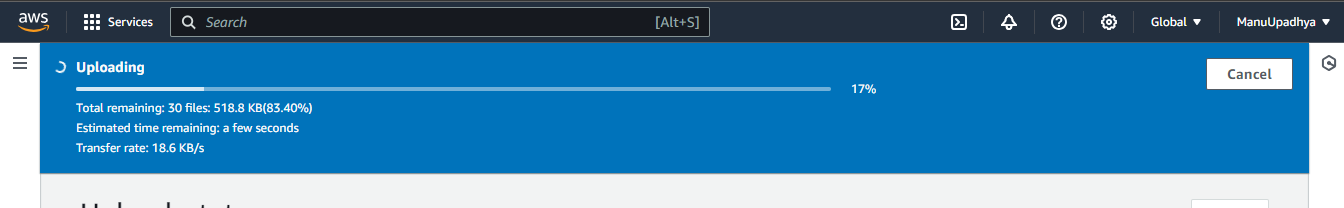
Keep **Server-Side Encryption** same as default, keep **additional checksum** as off.

****

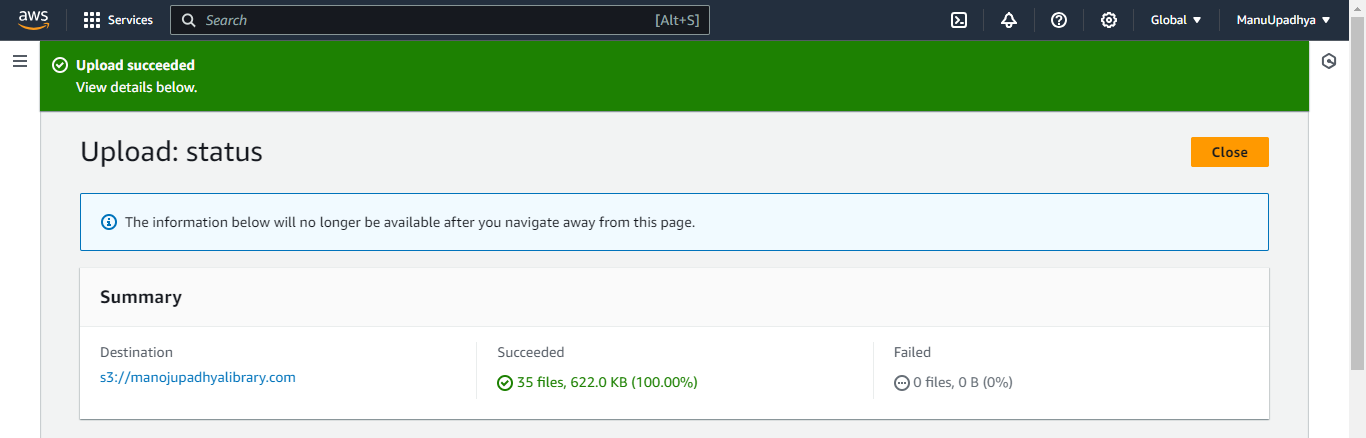
No “Tags” and “Meta data” are required.



Now click on “**Upload**” to Upload the website files as objects to S3 bucket.

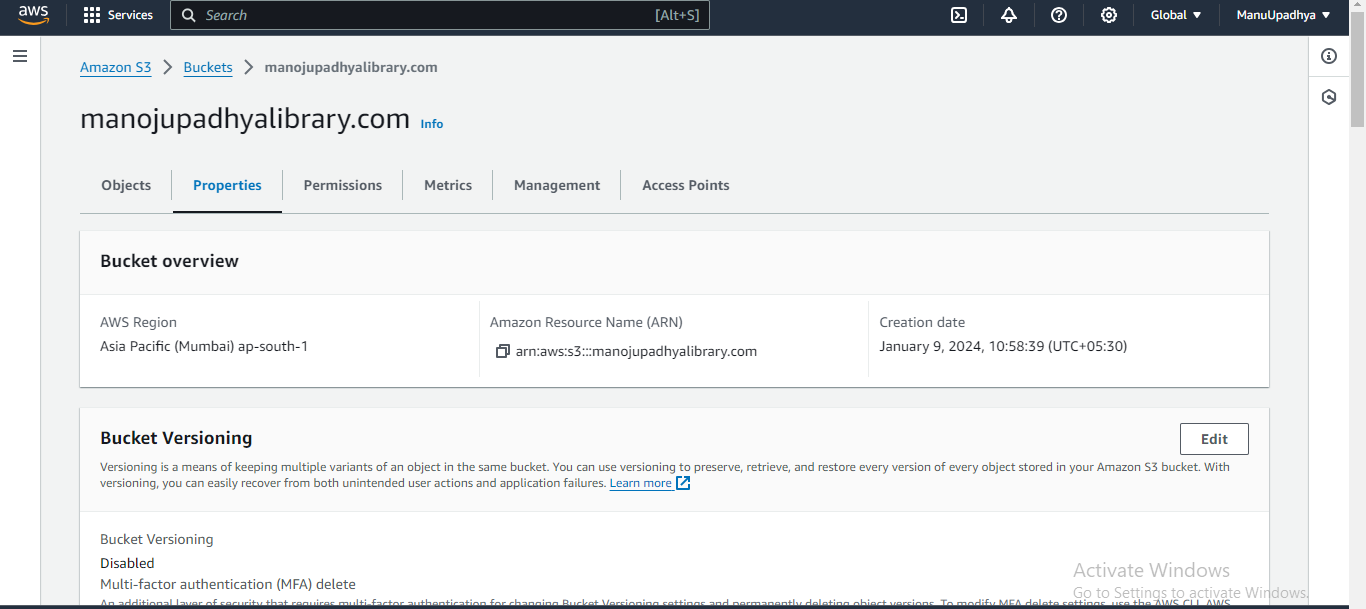


Once Uploading Completed Successfully. Make sure that bucket objects are public(Select all objects-> Actions -> make public using ACL).

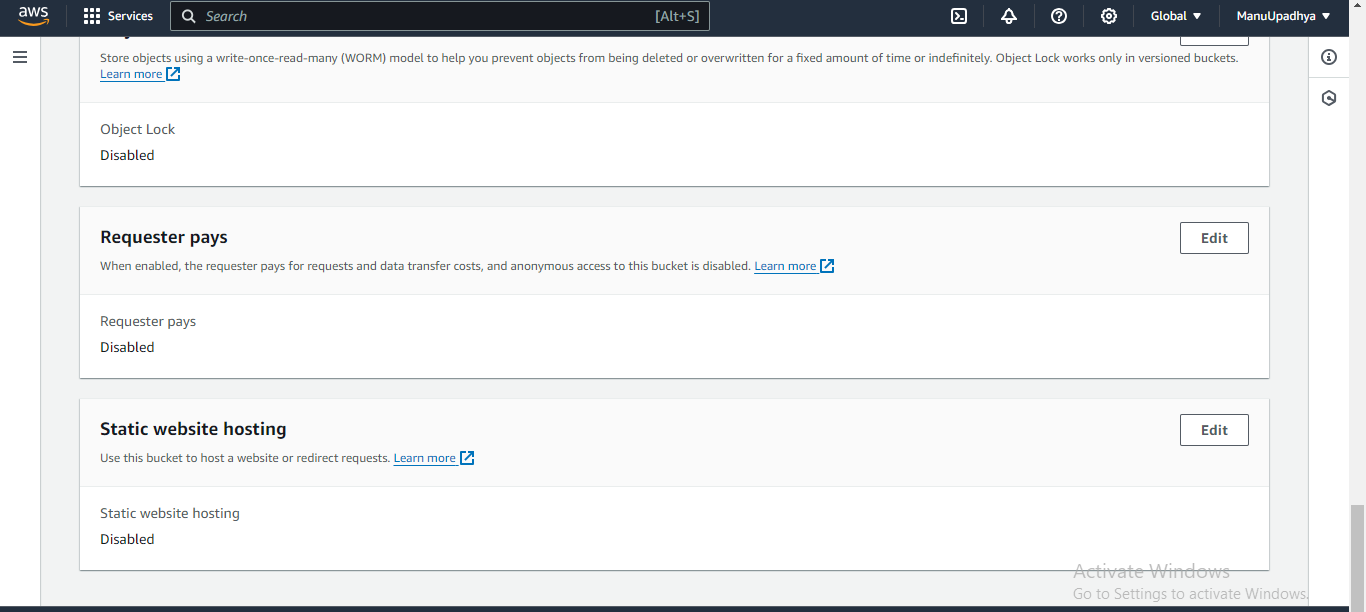


**Next step is Hosting our S3 buckets as a website** (enabling bucket objects to host as a website)

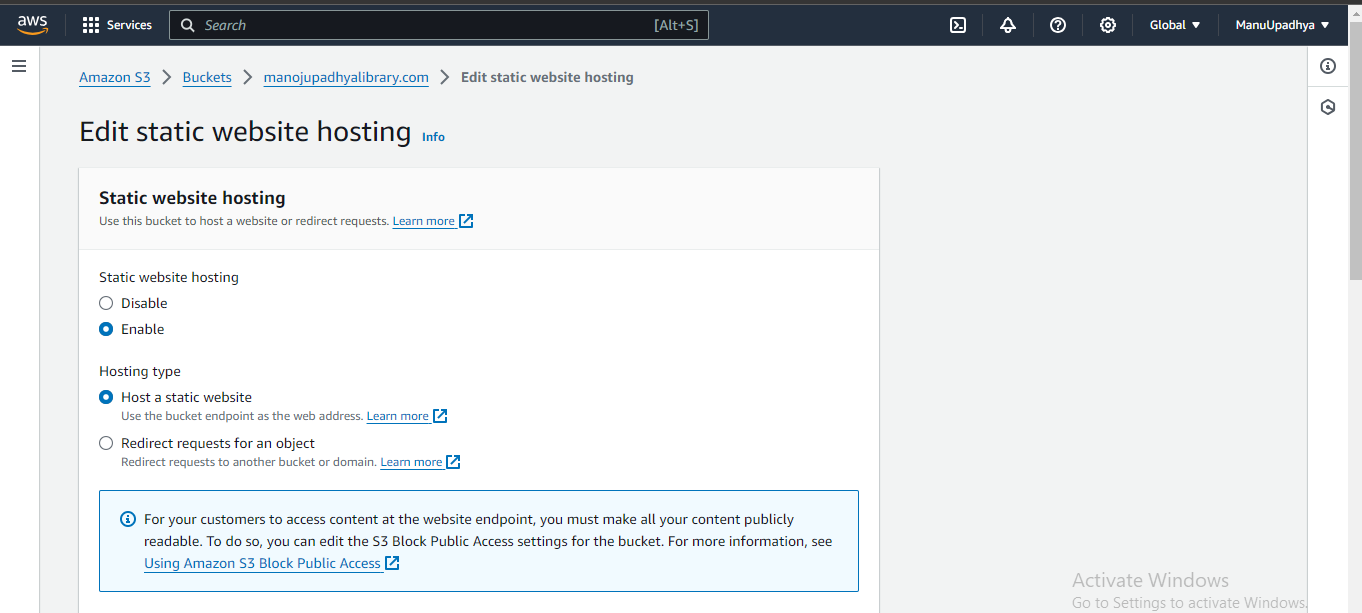
From inside the bucket navigate to properties.



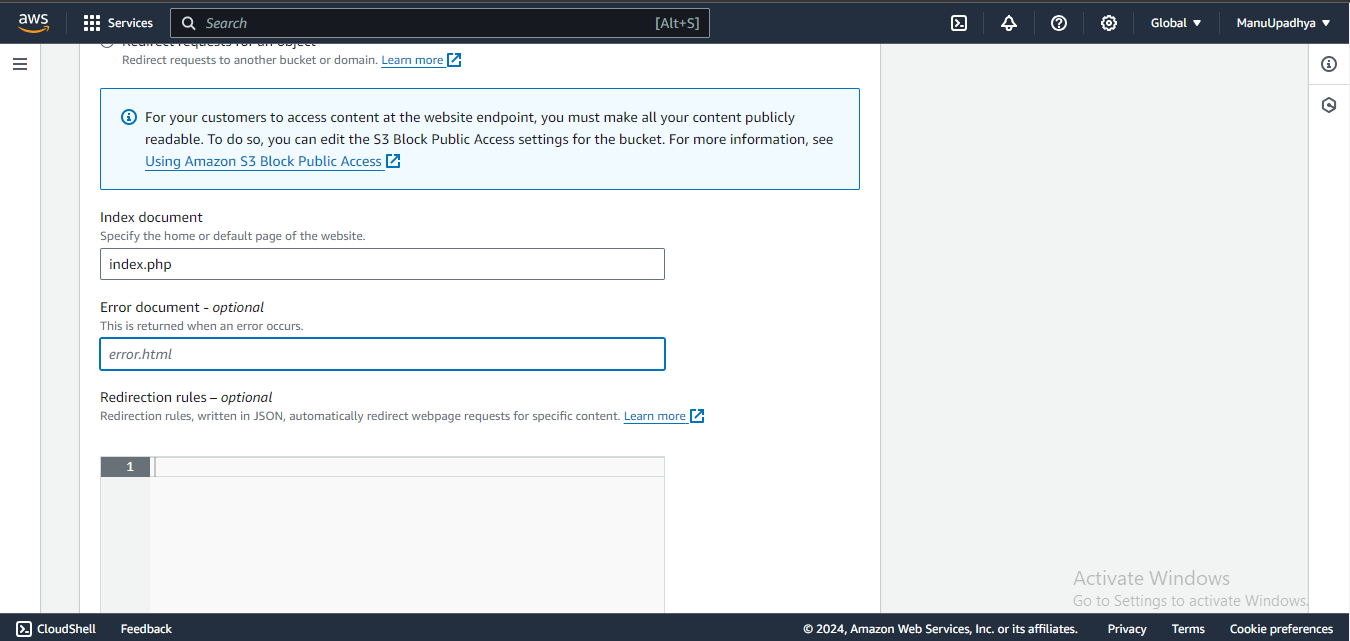
Navigate to “static Website hosting” .



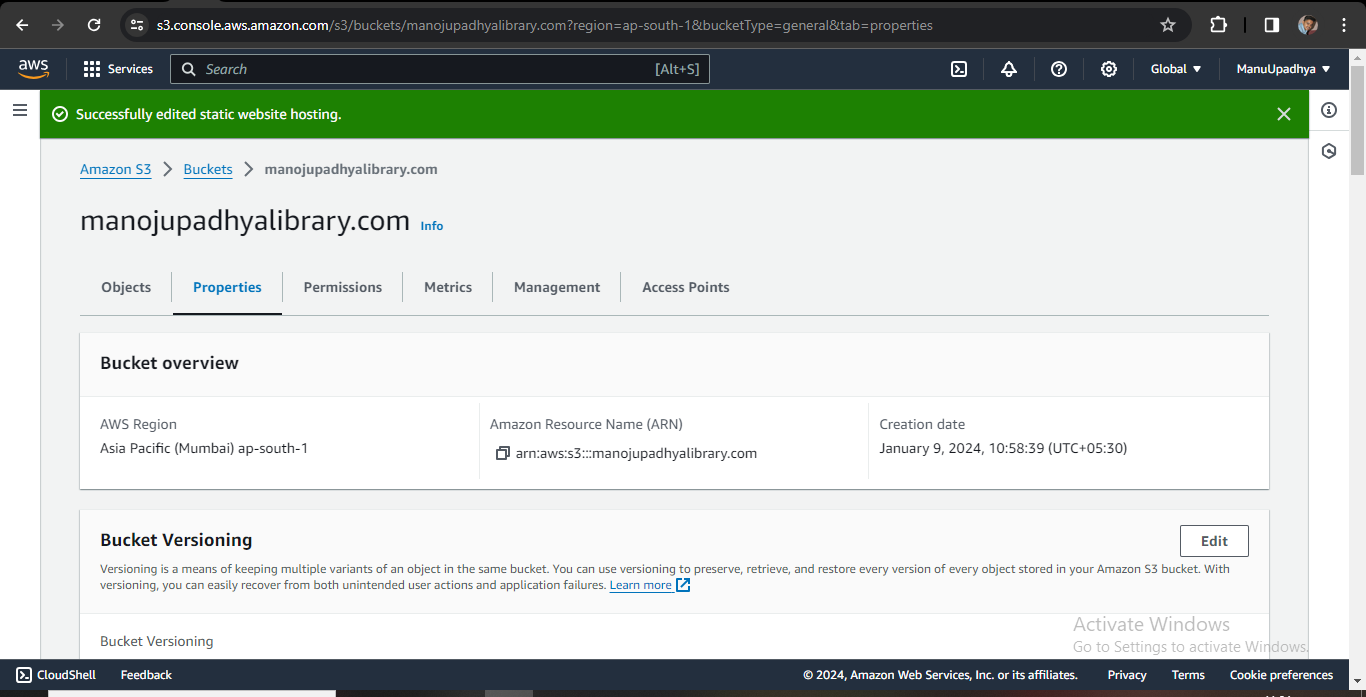
then click on “Edit” to enable the static website hosting option, by default it will be disabled.



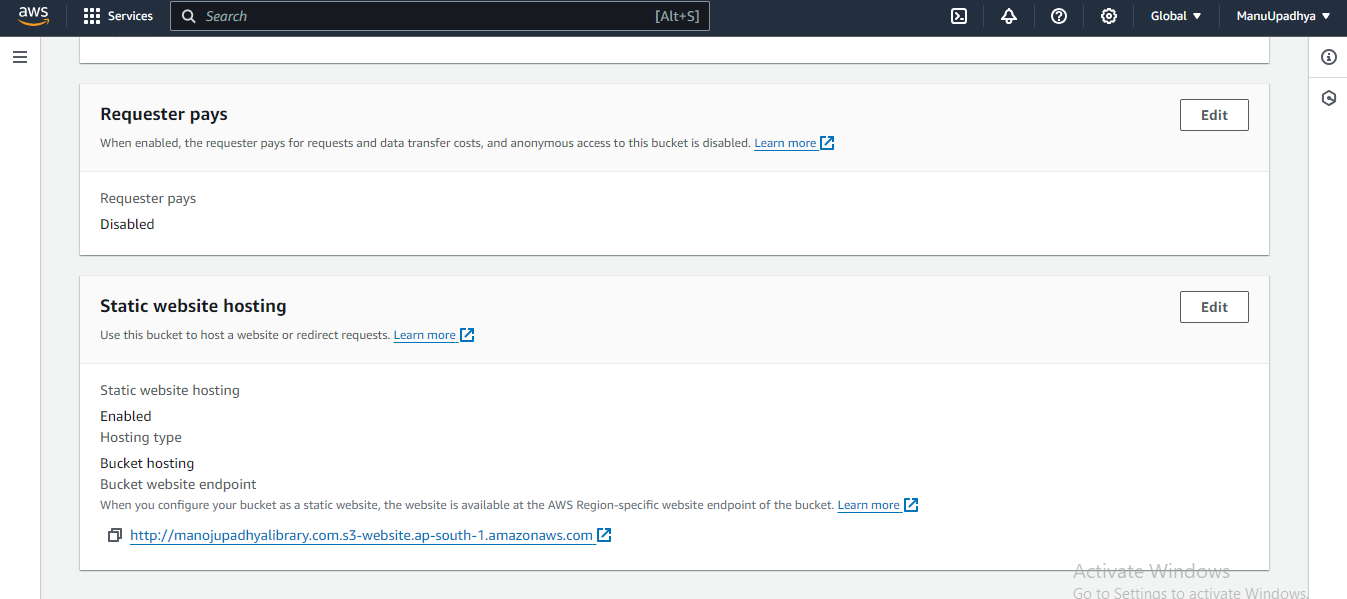
Insert index and error document name of our website.



Then click on “Save changes”. Successfully enabled the static website hosting.



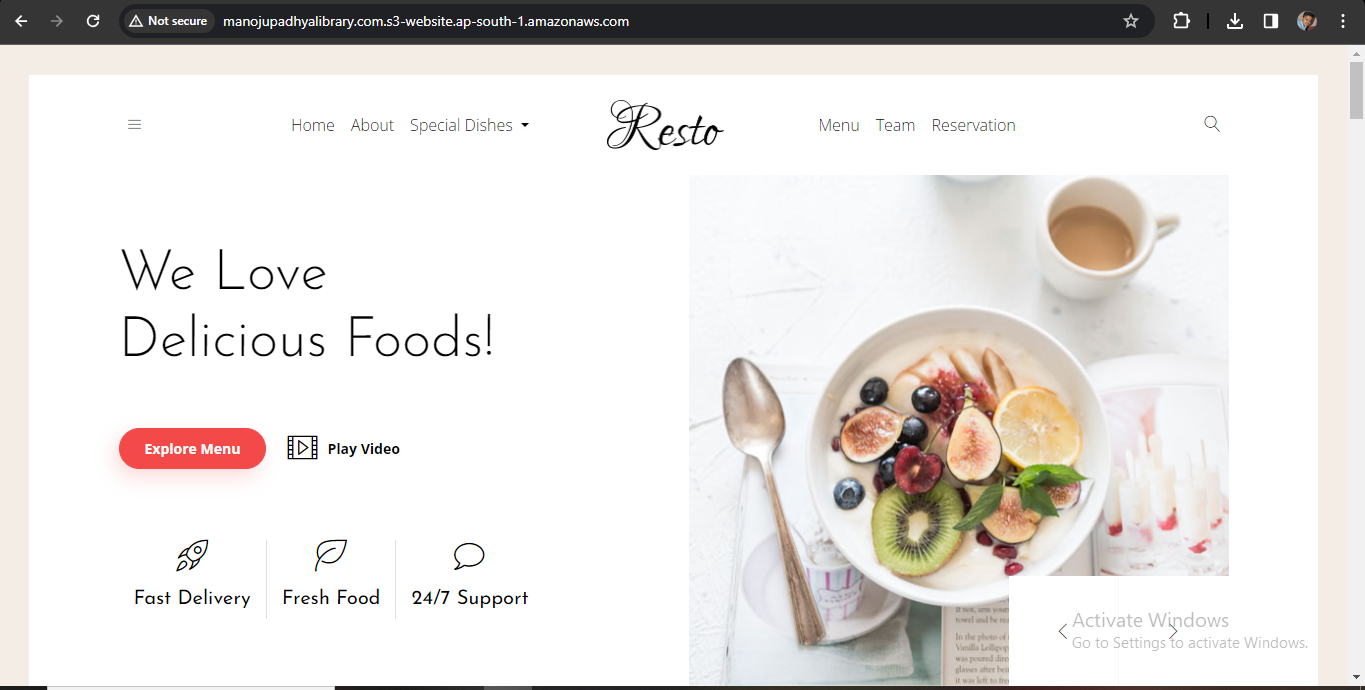
Now Navigate again to “Static website Hosting” option in Properties of Bucket to get the endpoint of our website.

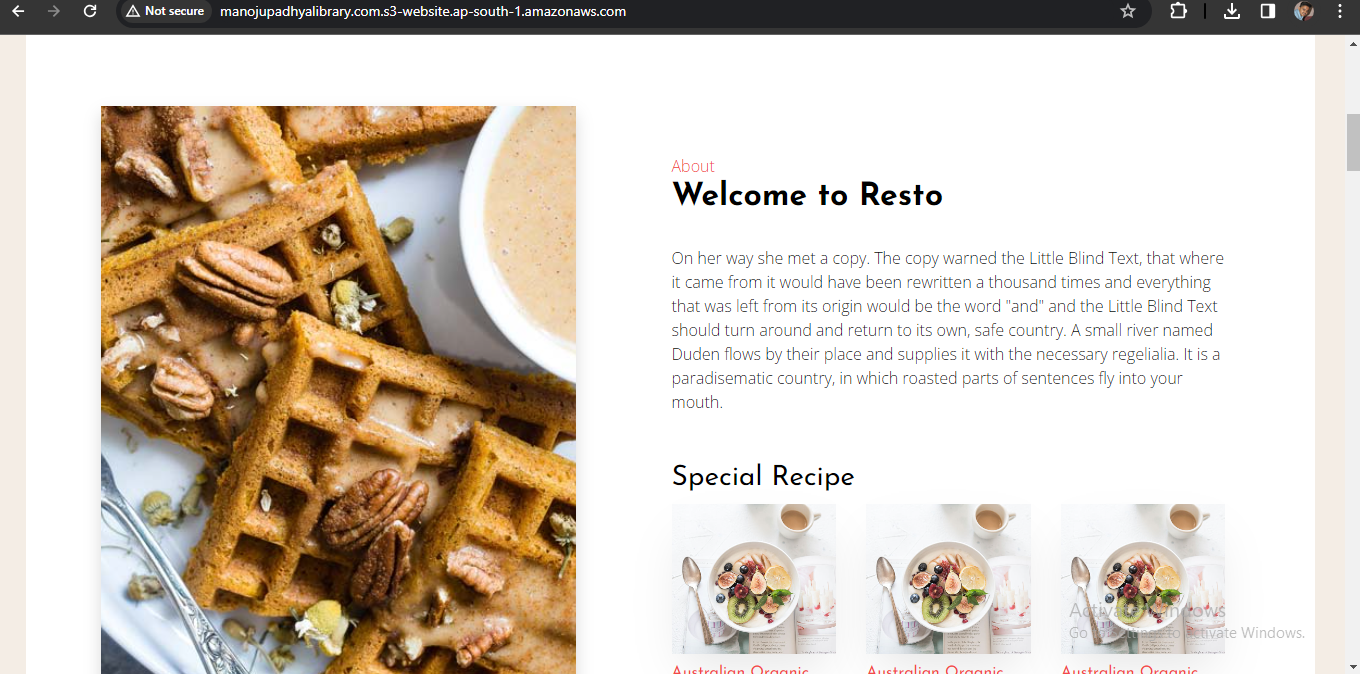


When we open the link/ endpoint we will see our hosted/static website

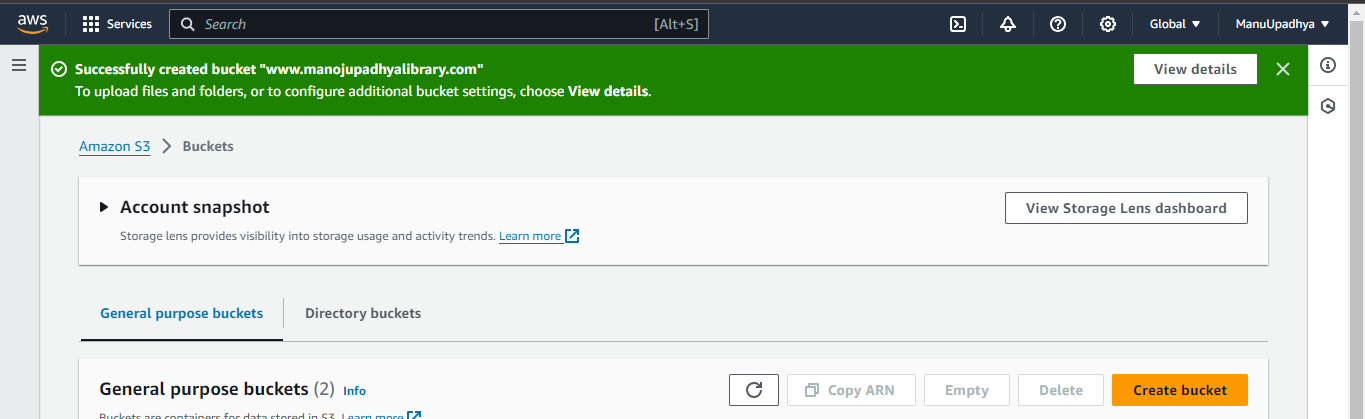
Link to our static website:-

http://manojupadhyalibrary.com.s3-website.ap-south-1.amazonaws.com





One problem here we can face is if we start our URL to site as www. manojupadhyalibrary.com.s3-website.ap-south-1.amazonaws.com It won’t land on our website; it will throw 404 error because www is not identified by our bucket. So, to resolve this we create one more bucket **for www. manojupadhyalibrary.com** and will route the traffic coming to this bucket to the original bucket **manojupadhyalibrary.com.**



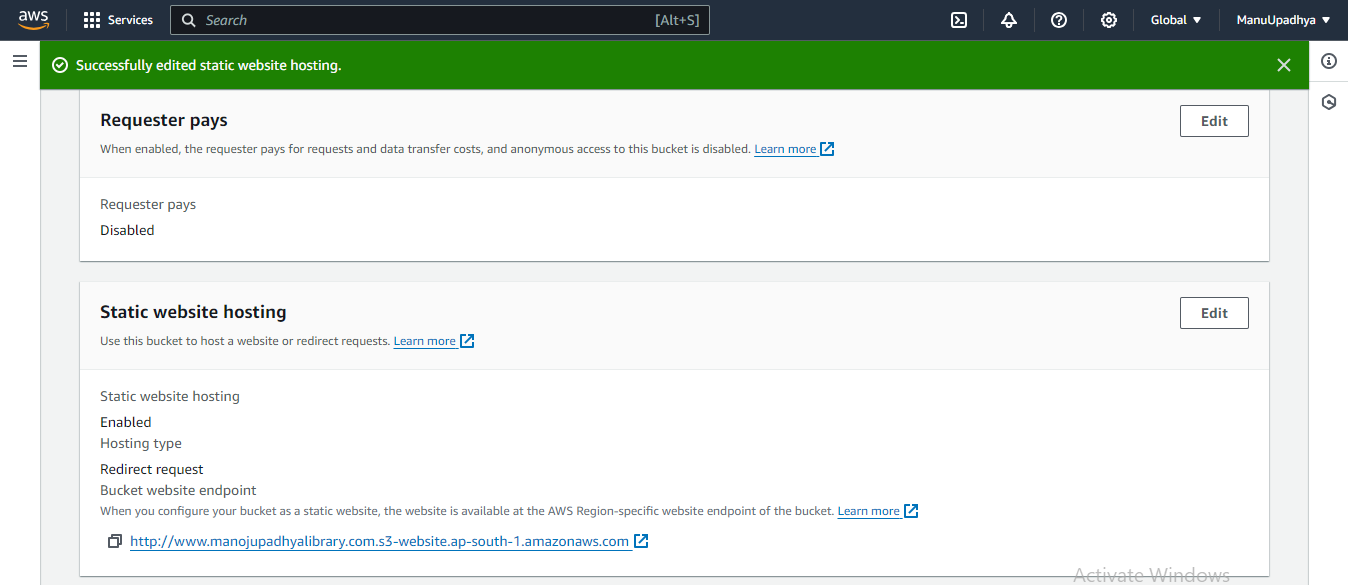
Now go inside the newly created bucket, navigate to properties then go to static website hosting.

Now choose “Redirect request for an object” and enter the target bucket name and click save changes.

A screenshot of a computer

Description automatically generated

Now we will go again to the second bucket which is created for **www. manojupadhyalibrary.com**, Navigate to properties then to Static website Hosting.



If we open the end point it will land to our static website.

A screenshot of a computer

Description automatically generated

1. **Creating SSL Certificate Using Certificate Manager**

Navigate to Services->Certificate Manager Management Console

A screenshot of a computer

Description automatically generated

Click on Request a Certificate-> then enter our domain name

A screenshot of a computer

Description automatically generated

**As I have not purchased the domain name it will throwing me error.**

We can add our subdomains as well, if we have multiple subdomains, we can use wild cards, and enter our sub domain name as **\*.domain name.**

Certificates need to be validated; link will be sent to our mail address. After approving the certificate, we will see the newly created certificate in certificate management console.

1. **Creating CloudFront Distribution**

Goto Services->CloudFront management console

A screenshot of a computer

Description automatically generated

Now create a distribution

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

If we want a Web application firewall, we can add it from here for the purpose of security.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Here we are informing to use all edge location for best performance.

A screenshot of a computer

Description automatically generated

Here we will select our Created Custom SSL Certificate.

A white background with black text

Description automatically generated

Here we will add domain name of our website.

Add root object that is index.html and Click on create distribution.

A screenshot of a computer

Description automatically generated

Created Distribution can be found in CloudFront distribution management console.

Initially the status will be deploying after some time status will be changed to deployed.

A screenshot of a computer

Description automatically generated

Navigate to invalidation tab in CloudFront Distribution. Create invalidation And add files which we need to invalidate. If we want to invalidate everything enter the wild card \*

A screenshot of a computer

Description automatically generated

1. **Routing traffic with AWS Route 53**

A screenshot of a computer

Description automatically generated

This is the CloudFront Distribution Domain name, if we open this in web browser it will directly land on our website. We will use this as alias target name in Route 53 for previously created domain name.

After successful setup if user enters just domain in web browser, he will be directed to alia target name that is CloudFront Distribution Domain name, and website will be loaded in his web browser.

**Not performed this step as I have not created Domain name. for my website. Still I can access the hosted website using the S3 bucket endpoint.**