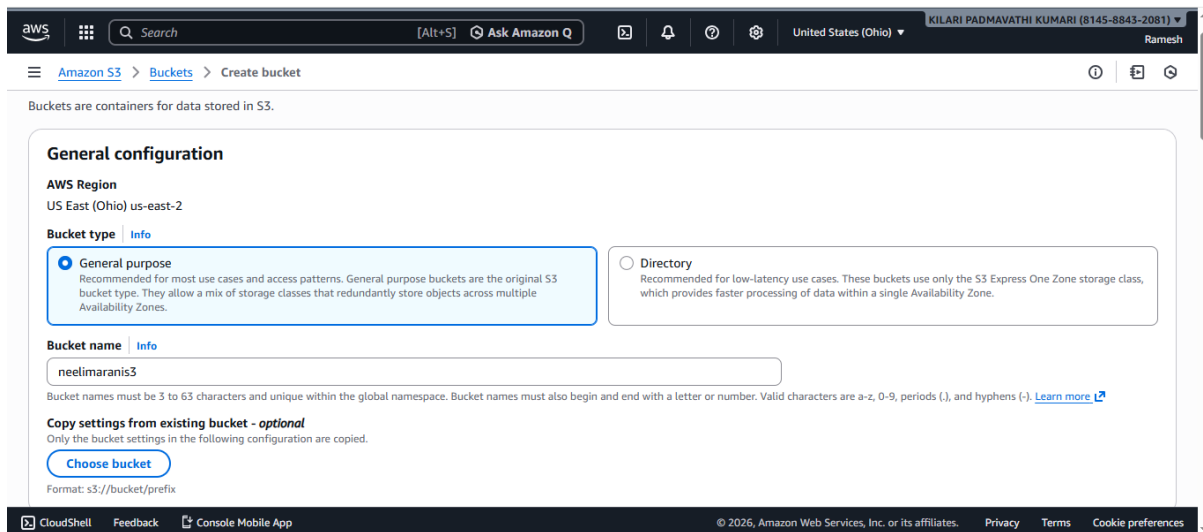


S3-TASK1

Amazon S3 (Simple Storage Service) is a highly scalable, durable, and secure cloud object storage service from Amazon Web Services (AWS).

1. Create an S3 bucket and upload some objects to S3.

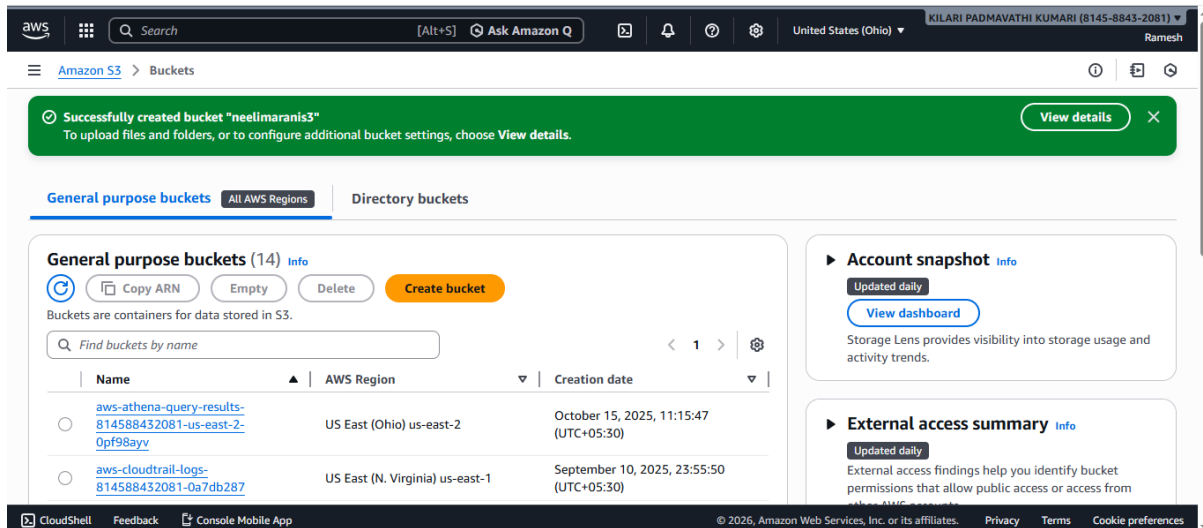
- Open Amazon S3
- Click on Bucket
- Click on Create bucket.



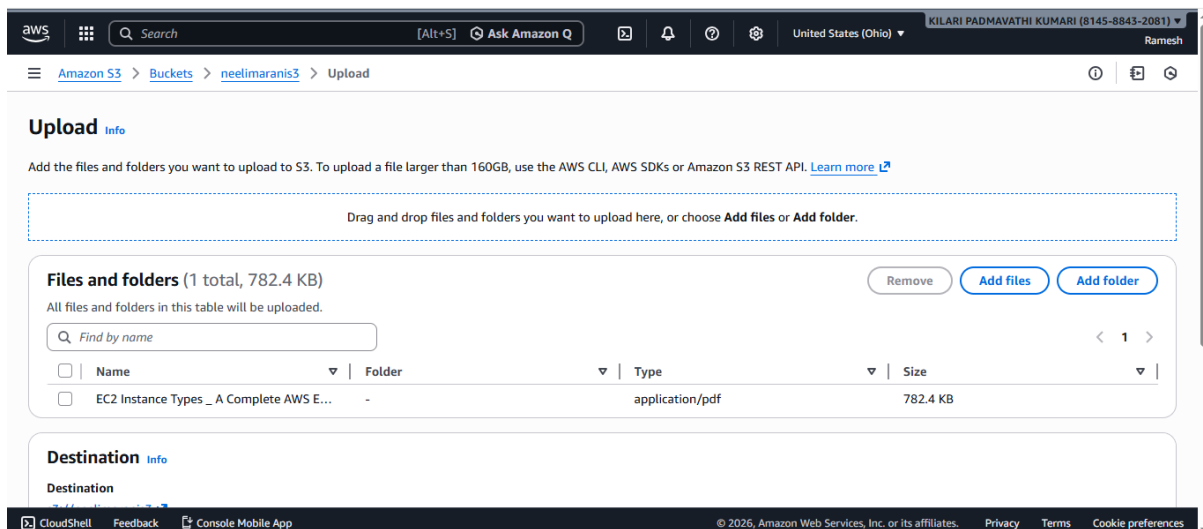
The screenshot shows the Amazon S3 'Create bucket' page in the AWS Management Console. The page is titled 'Create bucket' and includes a breadcrumb trail: 'Amazon S3 > Buckets > Create bucket'. Below the title, it states 'Buckets are containers for data stored in S3.' The 'General configuration' section is active, showing the 'AWS Region' as 'US East (Ohio) us-east-2'. Under 'Bucket type', the 'General purpose' option is selected, with a description: 'Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.' The 'Directory' option is also visible, with a description: 'Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.' The 'Bucket name' field is populated with 'neelamaranis3'. A note below the field states: 'Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). [Learn more](#)'. At the bottom of the configuration section, there is a 'Copy settings from existing bucket - optional' section with a 'Choose bucket' button. The footer of the console shows '© 2026, Amazon Web Services, Inc. or its affiliates.' and links for 'Privacy', 'Terms', and 'Cookie preferences'.

- General configuration
- AWS region what we have choose before
- Bucket type → general purpose
- Bucket Name → neelamaranis3 (Unique Name)
- Click on create bucket.

S3-TASK1



- The above image shows bucket created successfully.



- To upload objects into S3 bucket
- Go to buckets choose bucket which is already created
- Click on upload and
- Click on add files and browse from pc.

S3-TASK1

The screenshot displays the AWS S3 console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and user information. Below this, a green banner indicates a successful upload. The main content area shows the 'Files and folders' tab selected, displaying a table with one file uploaded successfully.

Upload succeeded
For more information, see the Files and folders table.

Destination	Succeeded	Failed
s3://neelamaranis3	1 file, 782.4 KB (100.00%)	0 files, 0 B (0%)

Files and folders | Configuration

Files and folders (1 total, 782.4 KB)

Find by name

Name	Folder	Type	Size	Status	Error
EC2 Instance Types_A Com...	-	application/pdf	782.4 KB	Succeeded	-

CloudShell | Feedback | Console Mobile App | © 2026, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences

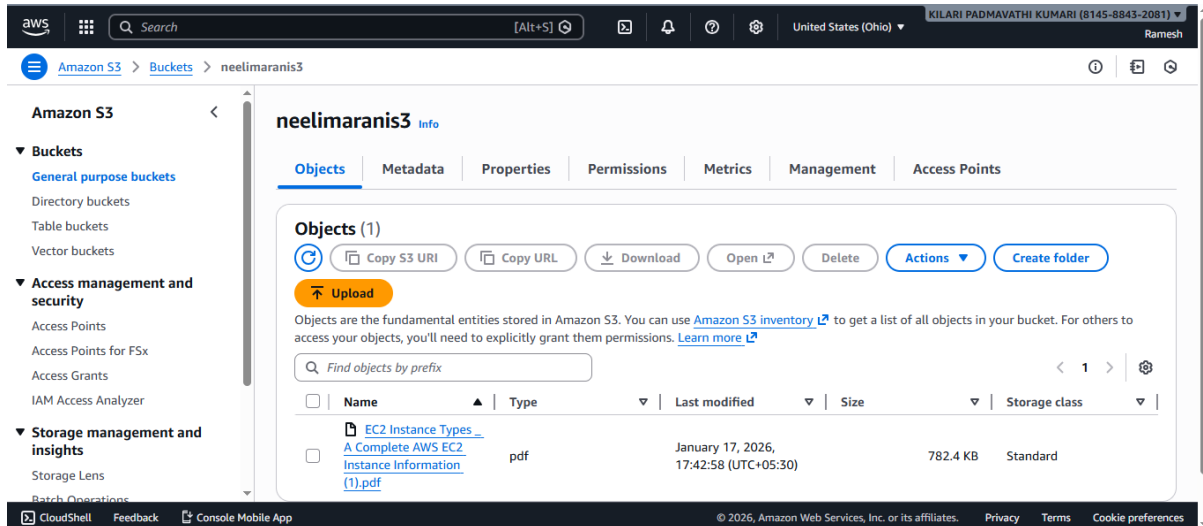
- The above image shows upload files successful and we can verify it.

S3-TASK1

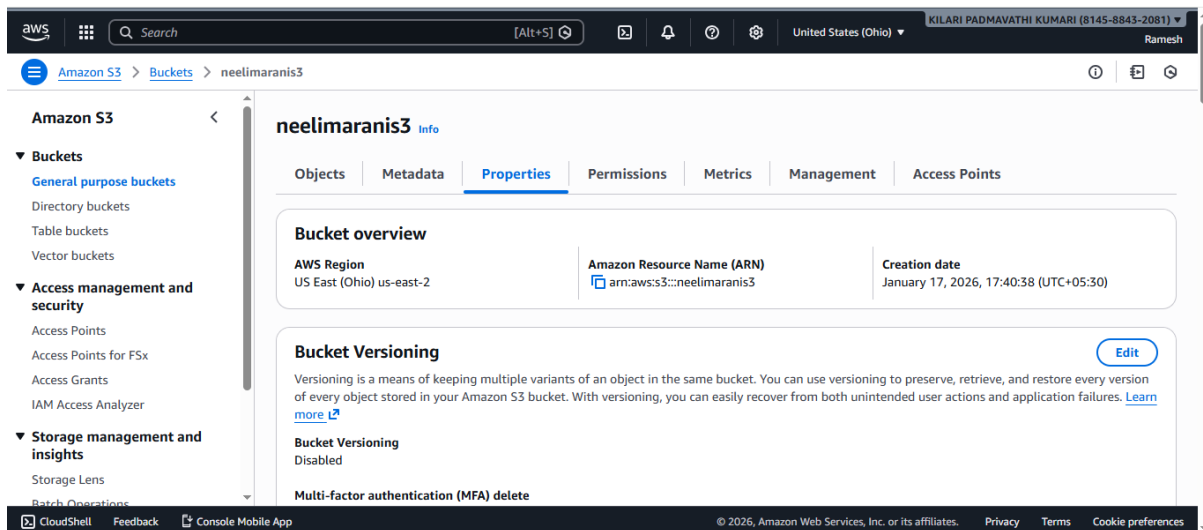
2. Deploy a static website in the S3 bucket.

Step 1: Enable Static Website Hosting

Go to AWS Console → S3 → Your Bucket



- Click **Properties** tab



S3-TASK1

- Scroll down to **Static website hosting** → Click **Edit**

The screenshot shows the AWS Management Console interface for the 'neelimarani3' bucket. On the left, the 'Amazon S3' navigation pane is visible, with 'Buckets' selected. The main content area shows the 'Static website hosting' settings. The 'Static website hosting' feature is currently disabled. A recommendation banner suggests using AWS Amplify Hosting for static website hosting. The 'Requester pays' feature is also disabled.

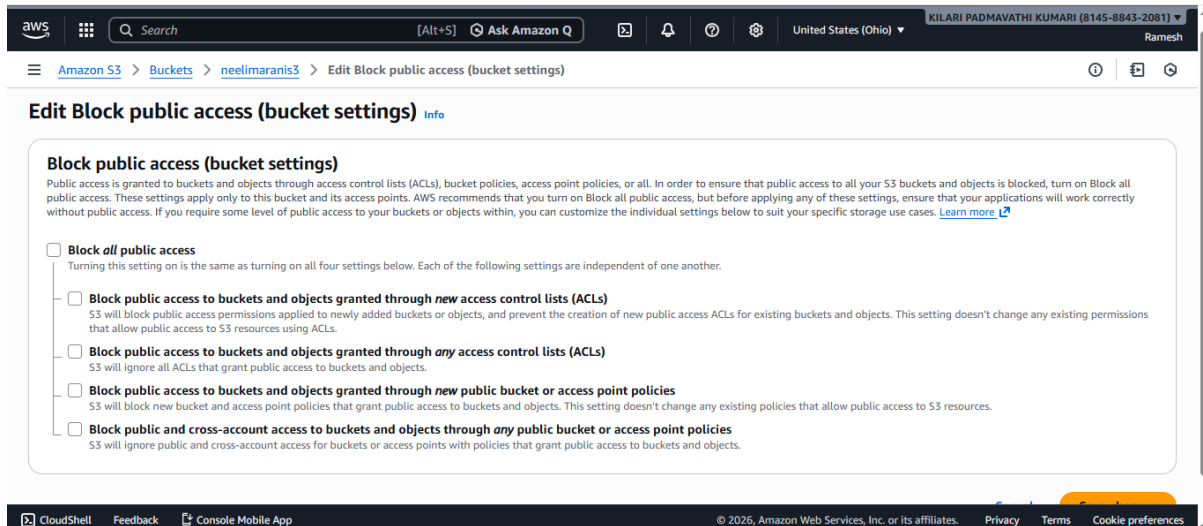
- Select **Enable**

The screenshot shows the 'Edit static website hosting' settings for the 'neelimarani3' bucket. The 'Static website hosting' feature is now enabled. The 'Hosting type' is set to 'Host a static website'. A note indicates that content must be publicly readable for the website endpoint.

The screenshot shows the 'Upload' page for the 'neelimarani3' bucket. The page displays a table of files and folders to be uploaded. The table shows two files: 'error.html' and 'login.html', both with a size of 193.0 B and 196.0 B respectively. The destination is set to 's3://neelimarani3'.

Name	Folder	Type	Size
error.html	-	text/html	193.0 B
login.html	-	text/html	196.0 B

S3-TASK1



- Save changes and confirm it.

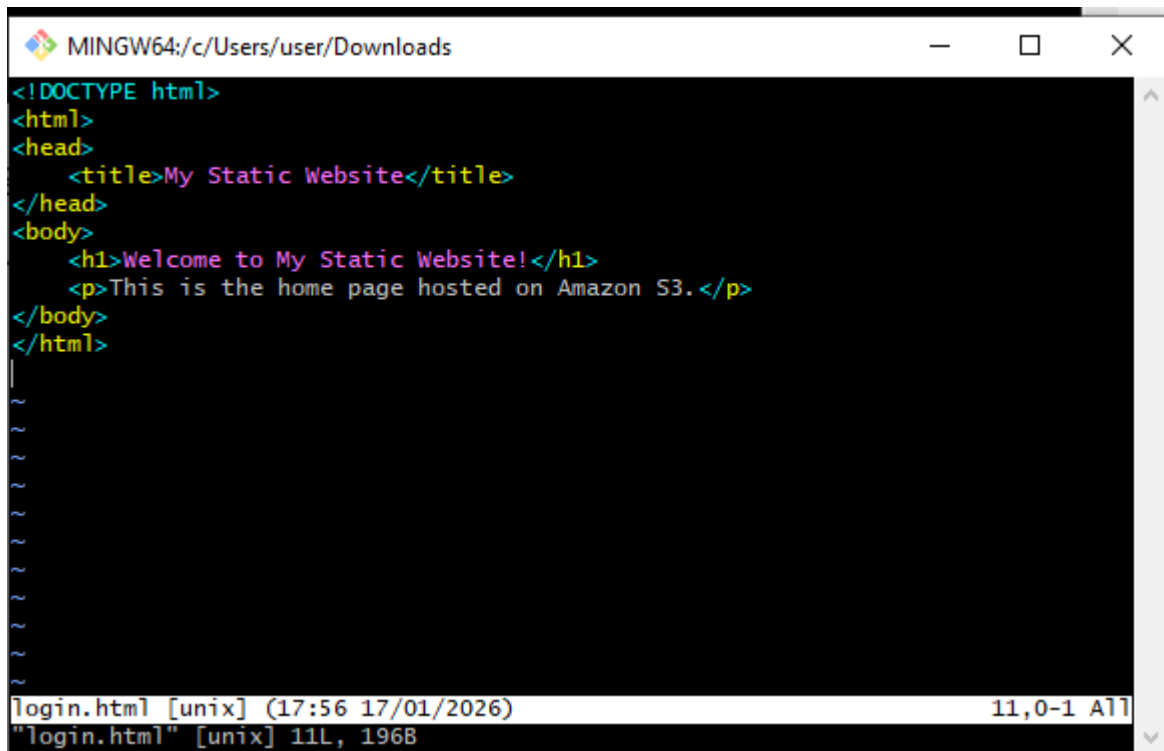
Choose **Host a static website**

Enter:

```
MINGW64:/c/Users/user/Downloads
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ vi login.html
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ vi error.html
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ |
```

- **Index document:** login.html

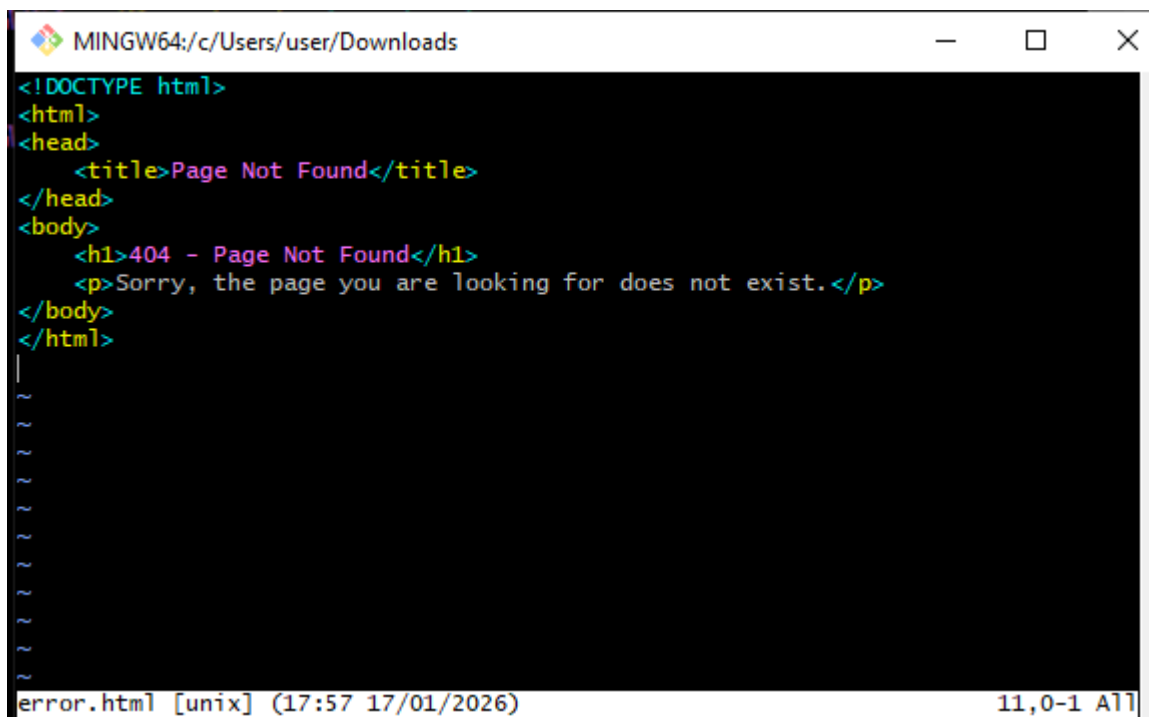
S3-TASK1



```
!DOCTYPE html>
<html>
<head>
  <title>My Static Website</title>
</head>
<body>
  <h1>Welcome to My Static Website!</h1>
  <p>This is the home page hosted on Amazon S3.</p>
</body>
</html>
```

login.html [unix] (17:56 17/01/2026) 11,0-1 All

- **Error document:** error.html (optional)



```
!DOCTYPE html>
<html>
<head>
  <title>Page Not Found</title>
</head>
<body>
  <h1>404 - Page Not Found</h1>
  <p>Sorry, the page you are looking for does not exist.</p>
</body>
</html>
```

error.html [unix] (17:57 17/01/2026) 11,0-1 All

S3-TASK1

The screenshot shows the AWS Management Console interface for uploading files to an S3 bucket. The breadcrumb navigation at the top indicates the path: Amazon S3 > Buckets > neelimaranis3 > Upload. The main area features a dashed box for dragging and dropping files, with instructions to 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.' Below this, a section titled 'Files and folders (2 total, 389.0 B)' contains a table of files to be uploaded. The table has columns for Name, Folder, Type, and Size. Two files are listed: 'error.html' (193.0 B) and 'login.html' (196.0 B). To the right of the table are buttons for 'Remove', 'Add files', and 'Add folder'. Below the table, the 'Destination' section shows the bucket name 'neelimaranis3' and a link to 's3://neelimaranis3'. At the bottom, there is a 'Destination details' section. The footer of the console includes links for CloudShell, Feedback, and Console Mobile App, along with copyright information for Amazon Web Services, Inc. and links for Privacy, Terms, and Cookie preferences.

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

Files and folders (2 total, 389.0 B) Remove Add files Add folder

All files and folders in this table will be uploaded.

<input type="checkbox"/>	Name	Folder	Type	Size
<input type="checkbox"/>	error.html	-	text/html	193.0 B
<input type="checkbox"/>	login.html	-	text/html	196.0 B

Destination [info](#)

Destination
[s3://neelimaranis3](#)

Destination details

- Click **Save changes**

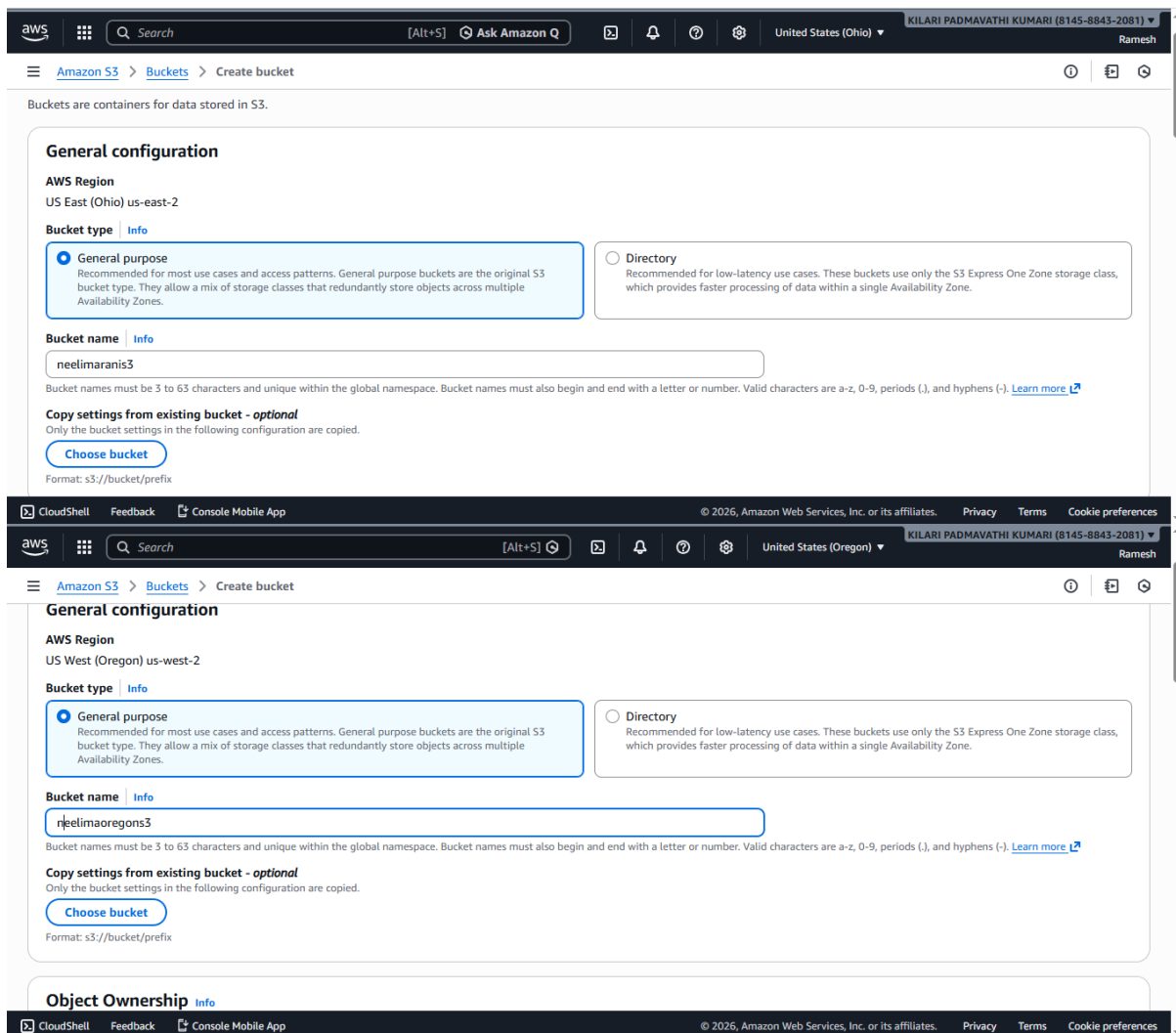
S3-TASK1

3. Enable cross-region replication on S3 buckets.

You need **two buckets**:

- Source bucket**
- Destination bucket** (in a different region)

Both buckets must have **versioning enabled** (CRR requires versioning).



The image displays two screenshots of the AWS Management Console's 'Create bucket' page, illustrating the configuration for two different buckets in different regions.

Top Screenshot (US East (Ohio) us-east-2):

- AWS Region:** US East (Ohio) us-east-2
- Bucket type:** General purpose (selected), Directory (unselected).
- Bucket name:** neelimaranis3
- Copy settings from existing bucket - optional:** Choose bucket (button).
- Format:** s3://bucket/prefix

Bottom Screenshot (US West (Oregon) us-west-2):

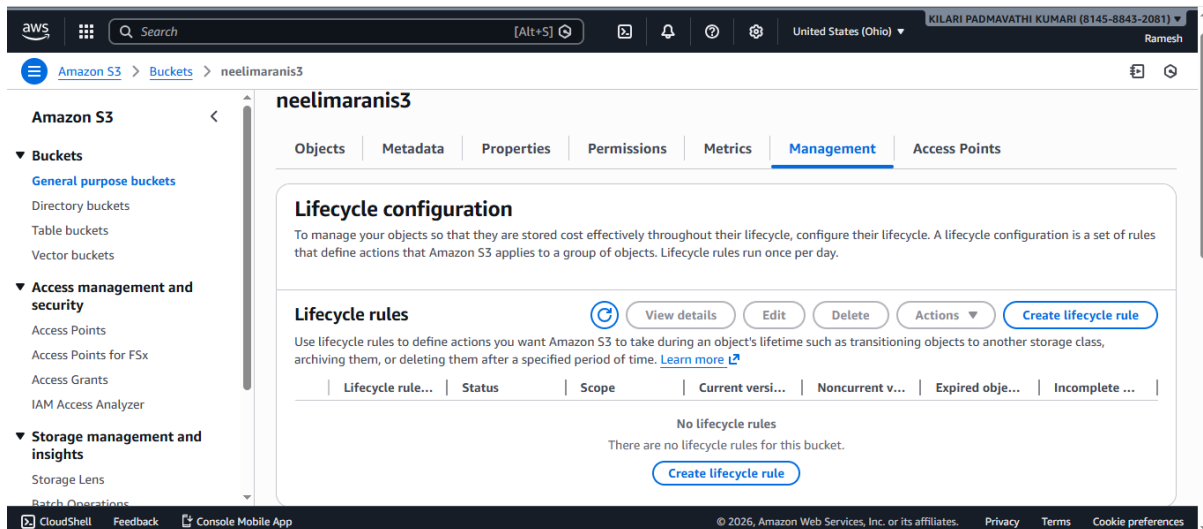
- AWS Region:** US West (Oregon) us-west-2
- Bucket type:** General purpose (selected), Directory (unselected).
- Bucket name:** neelimaoregions3
- Copy settings from existing bucket - optional:** Choose bucket (button).
- Format:** s3://bucket/prefix

Both screenshots show the 'General configuration' section with the 'General purpose' bucket type selected. The 'Bucket name' field is populated with a unique name for each region. The 'Copy settings from existing bucket - optional' section includes a 'Choose bucket' button. The 'Format' field shows the default S3 path format: s3://bucket/prefix.

S3-TASK1

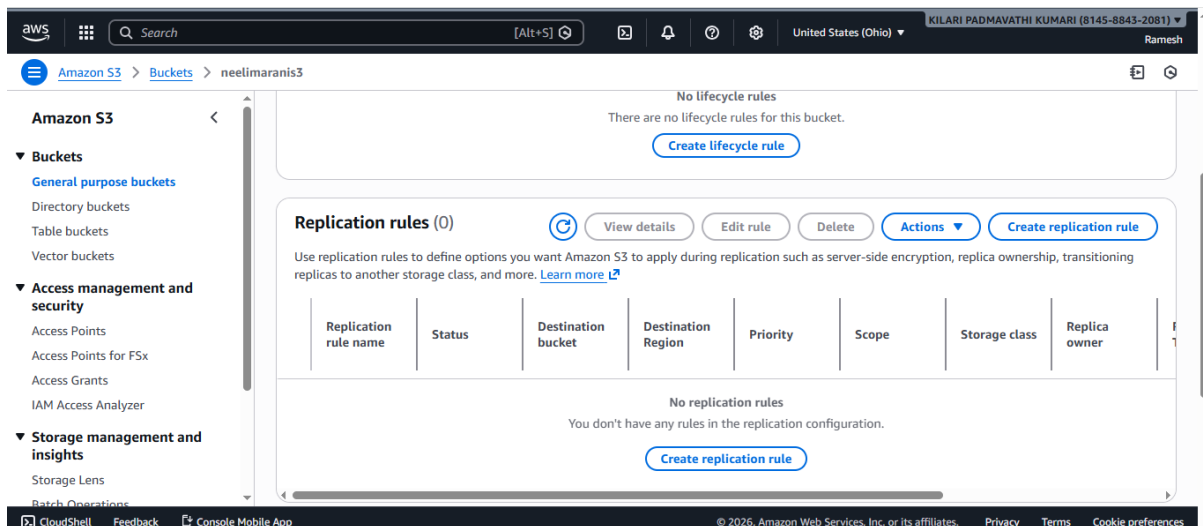
Step 1: Go to Source Bucket

- Log in to **AWS Console** → **S3** → **Select your source bucket** (e.g., neelimarani3)
- Click **Management** tab



Step 2: Create Replication Rule

- Scroll down to **Replication rules** → Click **Create rule**
- Enter a **Rule name**, e.g., CRR-to-oregon

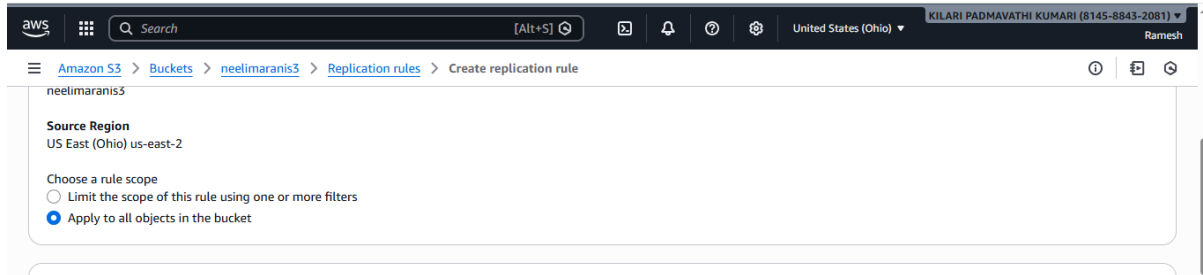


S3-TASK1

Step 3: Choose Rule Scope

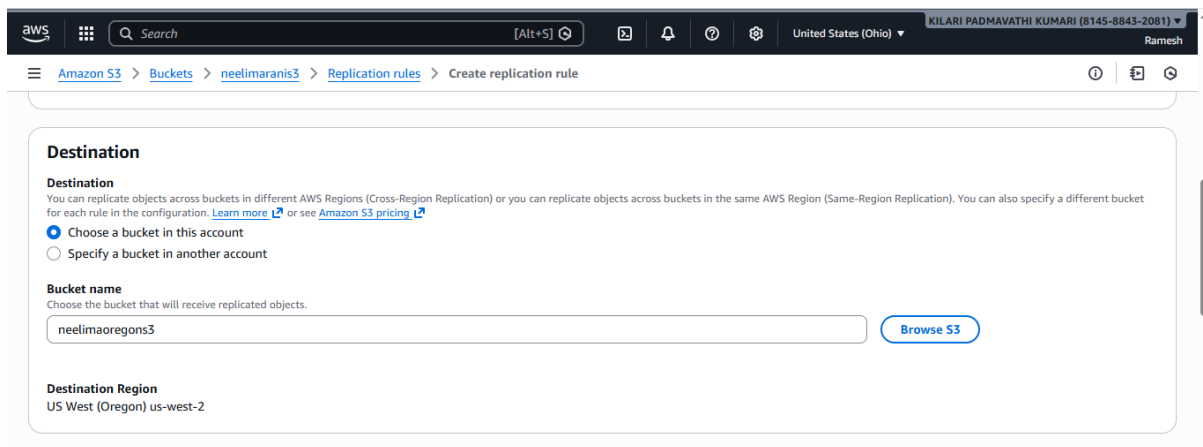
Rule scope:

- Select **Apply to all objects**
- Or choose **Filter by prefix or tags** if you only want certain objects replicated



Step 4: Choose Destination Bucket

- Destination → Select Another AWS Region
- Bucket → Select your destination bucket (e.g., neelimaoregons3-bucket)
- Storage class in destination → leave as Standard (or choose another like Glacier if needed)



S3-TASK1

Step 5: IAM Role for Replication

1. Choose or create a role:
 - Select Create a new role
 - AWS will create a role automatically (example: S3ReplicationRole)
 - This role allows S3 to replicate objects to the destination bucket
2. Click Save

Replicate existing objects?

You can enable a one-time Batch Operations job from this replication configuration to replicate objects that already exist in the bucket and to synchronize the source and destination buckets. [Learn more](#) or [see pricing](#)

Existing objects

☐ No, do not replicate existing objects.

☒ Yes, replicate existing objects.

[Cancel](#) [Submit](#)

The screenshot shows the AWS S3 console 'Upload' page for the bucket 'neelimaranis3'. The page includes a search bar, a list of files to be uploaded, and a destination field. The file 'IAM_task.pdf' is listed with a size of 3.3 MB. The destination is set to 's3://neelimaranis3'.

Name	Folder	Type	Size
IAM_task.pdf	-	application/pdf	3.3 MB

- Here we have upload files in source bucket nothing neelimaranis3.

S3-TASK1

aws [Search] [Alt+S] United States (Oregon) KILARI PADMAVATHI KUMARI (8145-8843-2081) Ramesh

Amazon S3 Buckets neelimaoregons3

neelimaoregons3 Info

Objects (1)

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](#)

Find objects by prefix Show versions

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	IAM_task.pdf	pdf	January 17, 2026, 18:42:09 (UTC+05:30)	3.3 MB	Standard

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Here it is reflected in neelimaoregons3 due to replication rule

S3-TASK1

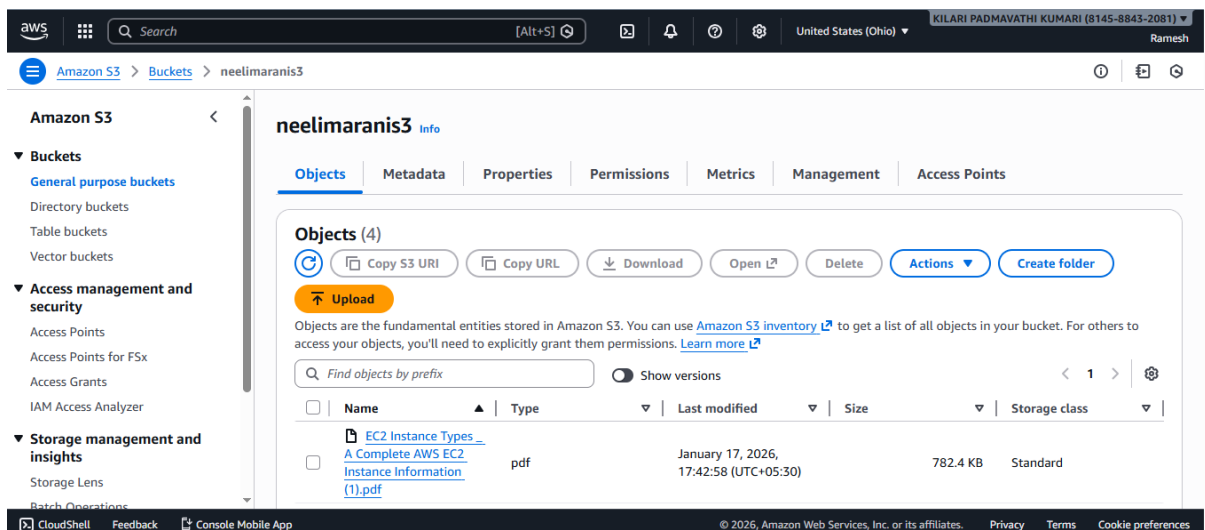
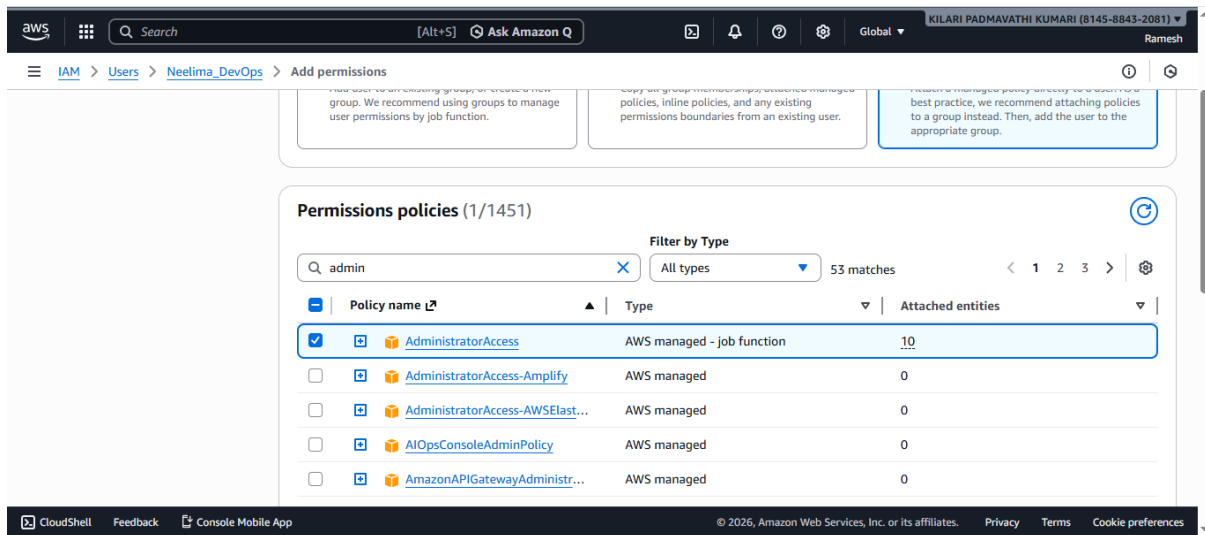
S3 Cross-Region Replication – Short Steps

1. Create 2 buckets
 - Source bucket → Region 1 (example: us-west-2)
 - Destination bucket → Region 2 (example: ap-south-1)
2. Enable Versioning
 - Go to S3 → Source bucket → Properties → Versioning → Enable
 - Go to S3 → Destination bucket → Properties → Versioning → Enable
3. Open Source Bucket
 - S3 → Source bucket → Management tab
4. Create Replication Rule
 - Click Replication rules → Create rule
 - Rule scope: Apply to all objects
5. Select Destination
 - Choose Another AWS Region
 - Select the destination bucket
6. IAM Role
 - Choose Create new role (AWS creates it automatically)
7. Create Rule
 - (Optional) Enable Replicate existing objects
 - Click Create rule
8. Verify
 - Upload a file to source bucket
 - File appears automatically in destination bucket

S3-TASK1

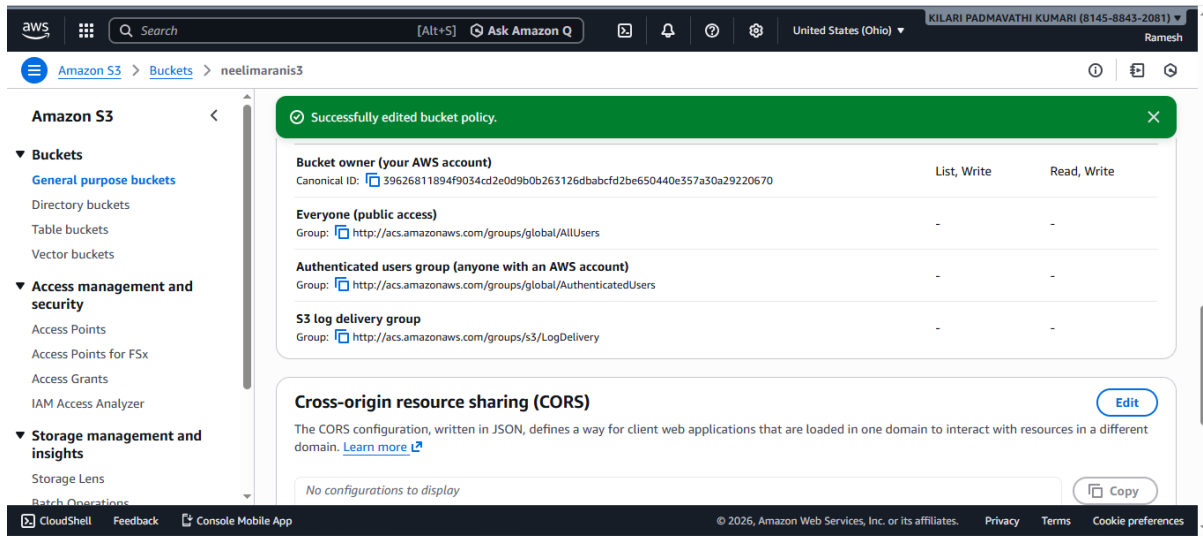
4. Configure a bucket policy so only the admin user can see the objects of the S3 bucket.

- In this admin user can see the objects means user should AdministratorAccess.
- IAM→user→Neelima_Devops→ Add permissions add AdministratorAccess.

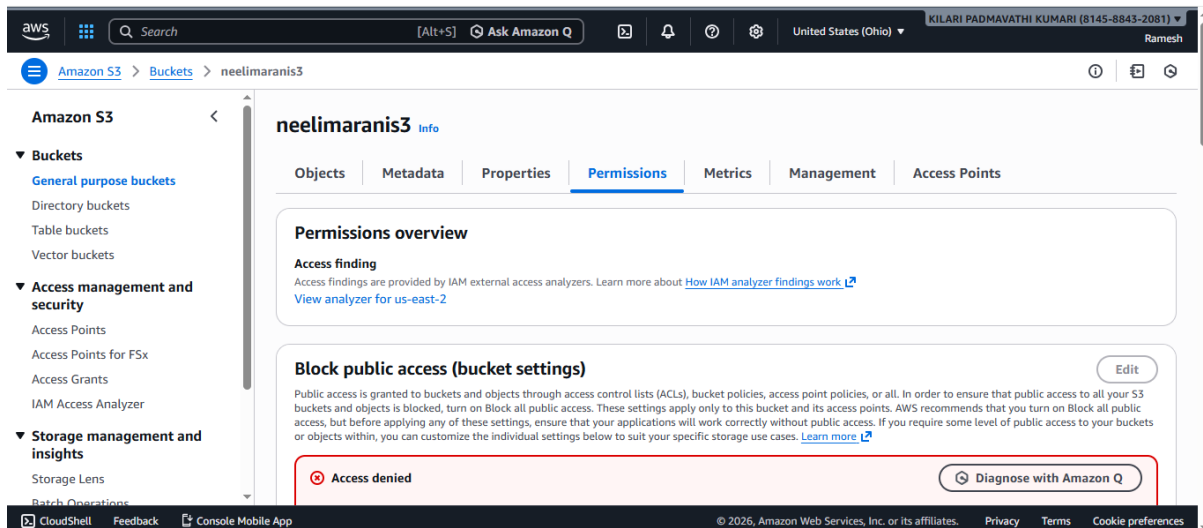


- Go to Amazon s3
- Click on buckets
- Select required bucket what we have created

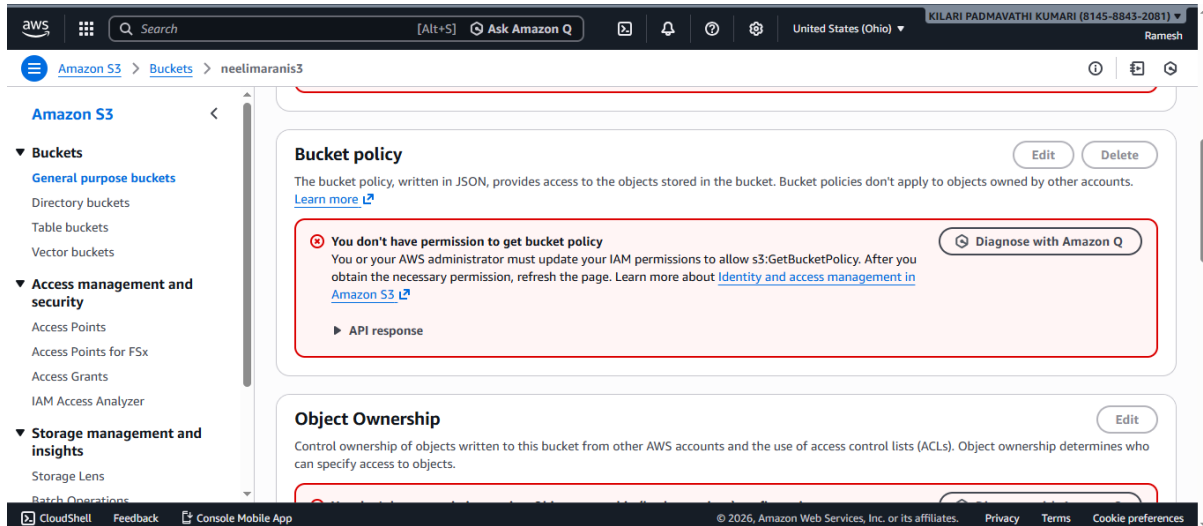
S3-TASK1



- Go to edit bucket policy and policy to it
- The above image shows successfully edit bucket policy.
- Now only admin user can access bucket .



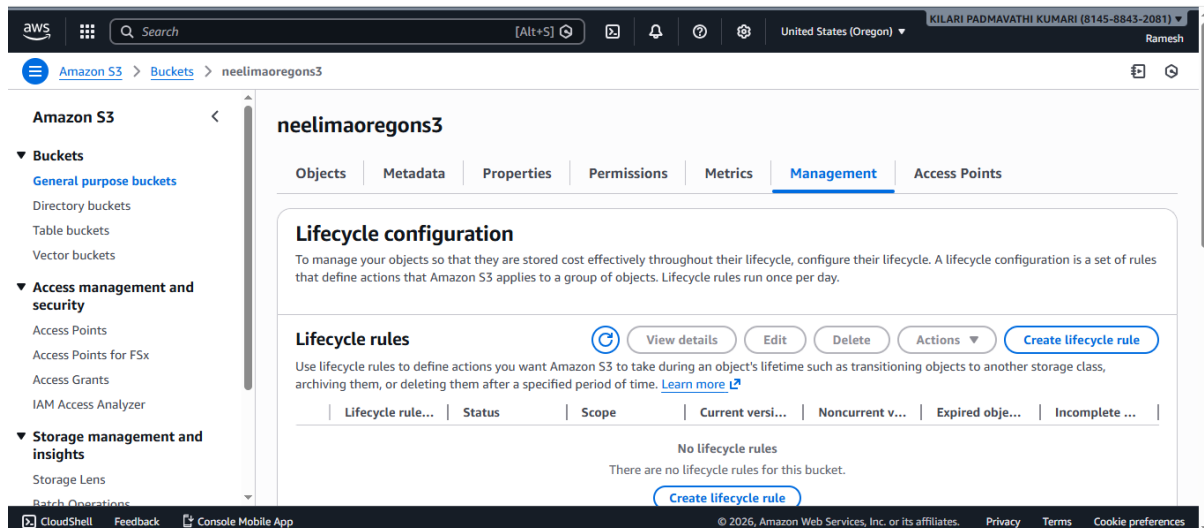
S3-TASK1



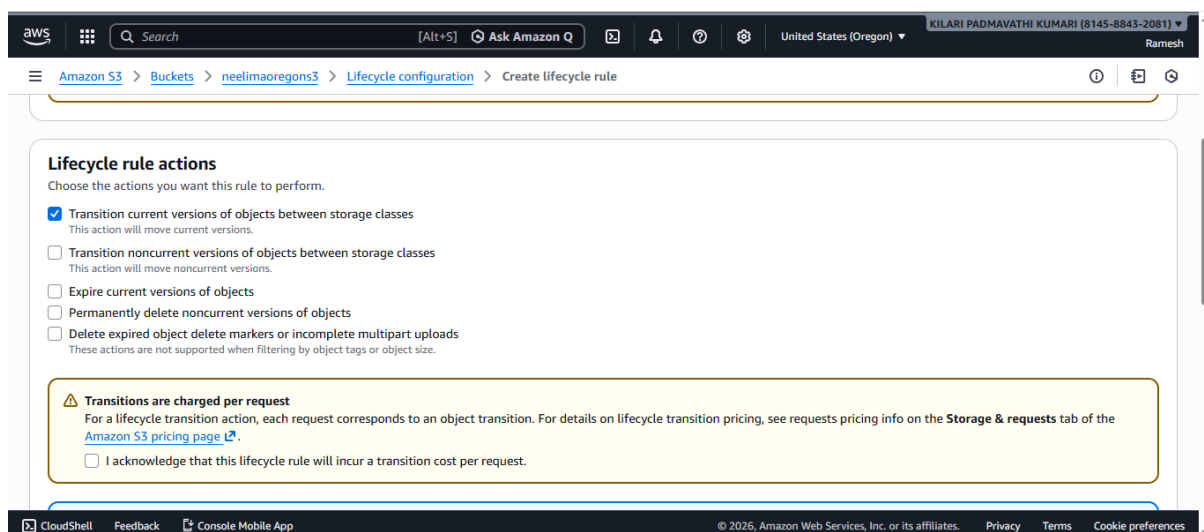
- To verify it other users can't access bucket and list the objects.

S3-TASK1

5. Set up lifecycle policies to automatically transition or delete objects based on specific criteria.



- Go to aws console and choose amazon s3
- Search for Buckets
- Choose bucket in which we want to make changes.
- After selecting bucket in that to go management tab
- Scroll and go to lifecycle configuration
- Click on create lifecycle rule.



- In Lifecycle rule actions choose the actions we want this rule to perform
- Select Transition current versions of objects b/w storage classes.

S3-TASK1

Transition current versions of objects between storage classes

Choose transitions to move current versions of objects between storage classes based on your use case scenario and performance access requirements. These transitions start from when the objects are created and are consecutively applied. [Learn more](#)

Choose storage class transitions

Glacier Flexible Retrieval (formerly Glacier) Days after object creation 30 Remove

Add transition

Review transition and expiration actions

Current version actions

Day 0

- Objects uploaded

↓

Noncurrent versions actions

Day 0

No actions defined.

- In Transition current version of objects between storage classes
- Choose storage class transitions → Glacier Flexible Retrieval
- Days after object creations → 30
- Click on create rule

Lifecycle configuration

To manage your objects so that they are stored cost effectively throughout their lifecycle, configure their lifecycle. A lifecycle configuration is a set of rules that define actions that Amazon S3 applies to a group of objects. Lifecycle rules run once per day.

Default minimum object size for transitions

All storage classes 128K

Lifecycle rules (1)

Use lifecycle rules to define actions you want Amazon S3 to take during an object's lifetime such as transitioning objects to another storage class, archiving them, or deleting them after a specified period of time. [Learn more](#)

Find lifecycle rules by name

Lifecycle rule na...	Status	Scope	Current version ...	Noncurrent vers...	Expired object d...	Incomplete mul...
Move-to-Glacier-30-days	Enabled	Entire bucket	Transition to Glacier Flex	-	-	-

- The above image shows Lifecycle configuration has been updated .

S3-TASK1

6. Push some objects to S3 using the AWS CLI.

MINGW64:/c/Users/user/Downloads

```
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ aws --version
aws-cli/2.32.26 Python/3.13.11 Windows/10 exe/AMD64

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ aws configure
AWS Access Key ID [*****MTIY]: AKIA33KKBB3I6K456467
AWS Secret Access Key [*****FtU5]: 5wTuFgg9XNqt6cEUFN4PDdaITFXP8QbysW
YeylPQ
Default region name [ap-south-1]: us-west-2
Default output format [json]: json

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ neelimaranis3
bash: neelimaranis3: command not found

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ aws sts get-caller-identity
{
  "UserId": "AIDA33KKBB3IWEUYMZHID",
  "Account": "814588432081",
  "Arn": "arn:aws:iam::814588432081:user/Neelima_DevOps"
}

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ |
```

- Open Git Bash check for **aws --version**
- Next configure aws with command **aws configure**
- After configure for details with command **Aws sts get-caller-identity**

```
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ vi Hello

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ aws s3 cp Hello s3://neelimaranis3/
upload: .\Hello to s3://neelimaranis3/Hello

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ |
```

- Create a File
- By using cp command we can push the objects into Buckets
Command: aws s3 cp Hello s3://neelimaranis3/

S3-TASK1

The screenshot shows the Amazon S3 console interface. On the left is a navigation sidebar with categories: Buckets, Access management and security, and Storage management and insights. The main panel is titled 'Objects (2)' and includes tabs for Objects, Metadata, Properties, Permissions, Metrics, Management, and Access Points. Below the tabs are buttons for Copy S3 URI, Copy URL, Download, Open, Delete, Actions, and Create folder. An 'Upload' button is also present. A text block explains that objects are fundamental entities stored in Amazon S3 and provides a link to the Amazon S3 inventory. Below this is a search bar and a 'Show versions' toggle. A table lists the objects in the bucket:

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	Hello	-	January 17, 2026, 19:55:22 (UTC+05:30)	16.0 B	Standard
<input type="checkbox"/>	IAM_task.pdf	pdf	January 17, 2026, 18:42:09 (UTC+05:30)	3.3 MB	Standard

The footer of the console shows links for CloudShell, Feedback, and Console Mobile App, along with the copyright notice: © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.

- To verify Go to aws console
- Choose s3 and go to bucket where we have pushed the object
- Check the objects by clicking on object tab.

S3-TASK1

7. Write a Bash script to create an S3 bucket.

```
MINGW64:/c/Users/user/Downloads
#!/bin/bash
BUCKET_NAME="neelamaranis3-demo-$(date +%s)"
REGION="us-west-2"

echo "creating s3 bucket: $BUCKET_NAME IN $REGION"

aws s3api create-bucket --bucket $BUCKET_NAME --region $REGION --create-bucket-configuration LocationConstraint=$REGION

if [ $? -eq 0 ]; then
    echo "Bucket created successfully"
else
    echo "Bucket creation failed"
fi
```

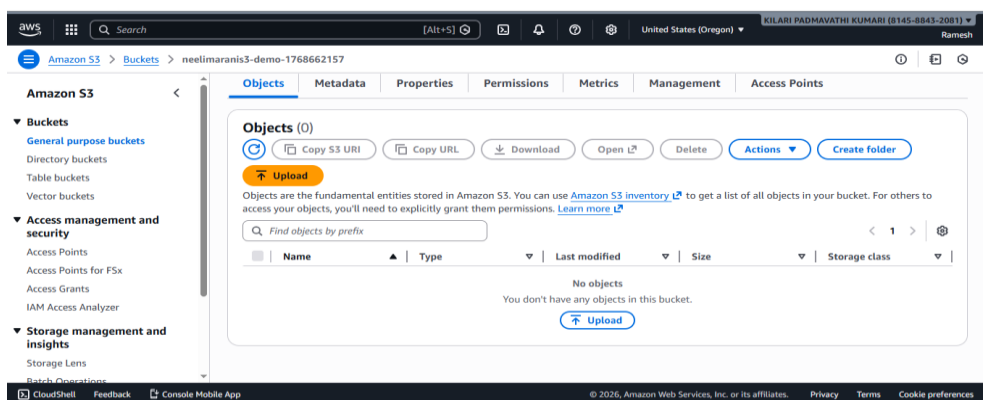
```
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ vi s3bucket.sh

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ chmod 777 s3bucket.sh

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ ./s3bucket.sh
creating s3 bucket: neelamaranis3-demo-1768662157 IN us-west-2
{
  "Location": "http://neelamaranis3-demo-1768662157.s3.amazonaws.com/",
  "BucketArn": "arn:aws:s3:::neelamaranis3-demo-1768662157"
}
Bucket created successfully

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ |
```

- Write Bash script by using vi editor.
- Create a file and write script and save it.
- Give execute permission to file
- Run by command ./filename.



- To verify to s3 service and check buckets, in that we can see newly created bucket through CLI.

S3-TASK1

8. Upload a 1 GB file to S3 using the CLI.

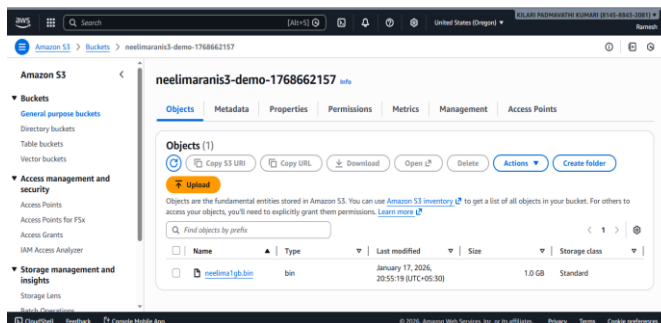
```
user@DESKTOP-3KH1RE MINGW64 ~/Downloads (master)
$ dd if=/dev/zero of=neelima1gb.bin bs=1M count=1024
1024+0 records in
1024+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 3.14687 s, 341 MB/s

user@DESKTOP-3KH1RE MINGW64 ~/Downloads (master)
$ ls -lh neelima1gb.bin
-rw-r--r-- 1 user 197121 1.0G Jan 17 20:53 neelima1gb.bin

user@DESKTOP-3KH1RE MINGW64 ~/Downloads (master)
$ aws s3 cp neelima1gb.bin s3://neelimaranis3-demo-1768662157/
upload: .\neelima1gb.bin to s3://neelimaranis3-demo-1768662157/neelima1gb.bin

user@DESKTOP-3KH1RE MINGW64 ~/Downloads (master)
$ |
```

- To upload 1GB files we have command
\$ dd if=/dev/zero of=neelima1gb.bin bs=1M count=1024
- **dd**: The command-line utility used for converting and copying files.
- **if=/dev/zero**: Defines the input file (if). /dev/zero is a special file in Unix-like systems that provides as many null characters (zero-value bytes) as are read from it.
- **of=neelima1gb.bin**: Defines the output file (of). This is the name of the file that will be created.
- **bs=1M**: Sets the block size (bs) to 1 Megabyte. This tells dd to read/write 1MB at a time.
- **count=1024**: Specifies that the command should only copy 1024 blocks.
- Check with command `ls -lh neelima1gb.bin`
- **aws s3 cp neelima1gb.bin s3://neelimaranis3-demo-1768662157/** → copy file to bucket.



- To verify, go to aws console choose buckets and select required bucket and check for objects tab and we can see file what we have create over the CLI.