**S3 Service CLI**

Follow Below are the steps along with the command syntax for performing various operations on Amazon S3 using the AWS Command Line Interface (CLI):

1. Create a Bucket:

To create an S3 bucket, you can use the aws s3 command create-bucket command:

**aws s3 mb s3://bucket-name**

Replace BUCKET\_NAME with your desired bucket name and REGION with the AWS region where you want to create the bucket.

2. Remove a Bucket:

To remove an S3 bucket, you can use the aws s3 command delete-bucket command:

**aws s3 rb s3://bucket-name**

Replace BUCKET\_NAME with the name of the bucket you want to delete. Note that the bucket must be empty before you can delete it.

3. Upload a File to S3 Bucket from Local:

To upload a file from your local machine to an S3 bucket, you can use the aws s3 cp command:

**aws s3 cp LOCAL\_FILE\_PATH s3://BUCKET\_NAME/DESTINATION\_PATH/**

4. Download a File from S3 Bucket to Local:

To download a file from an S3 bucket to your local machine, you can use the aws s3 cp command again, but reverse the source and destination:

**aws s3 cp s3://BUCKET\_NAME/SOURCE\_PATH/ LOCAL\_FILE\_PATH**

example

**aws s3 cp s3://bucket/filename ./**

Replace BUCKET\_NAME with the name of the source bucket, SOURCE\_PATH with the path to the file within the bucket, and LOCAL\_FILE\_PATH with the local destination path where you want to save the downloaded file.

5. Copy a File from One S3 Bucket to Another:

To copy a file from one S3 bucket to another, you can use the aws s3 cp command with source and destination paths:

**aws s3 cp s3://SOURCE\_BUCKET/SOURCE\_PATH/ s3://DESTINATION\_BUCKET/DESTINATION\_PATH/**

Replace SOURCE\_BUCKET and SOURCE\_PATH with the source bucket and path, and DESTINATION\_BUCKET and DESTINATION\_PATH with the destination bucket and path.

These AWS CLI commands should help you perform common S3 operations from the command line. Make sure to configure your AWS CLI with the necessary credentials and region before using these commands.

6. To move a file from one location to another within the same S3 bucket, you can use the aws s3 mv command. Here's the command syntax:

**aws s3 mv s3://BUCKET\_NAME/SOURCE\_PATH/ s3://BUCKET\_NAME/DESTINATION\_PATH/**

Replace BUCKET\_NAME with the name of the S3 bucket, SOURCE\_PATH with the path to the source file within the bucket, and DESTINATION\_PATH with the new path within the same bucket where you want to move the file.

This command will effectively move the file from the source location to the destination location within the same bucket.

7. List of buckets:

**aws s3 ls**

bucket details

**aws s3 ls s3://bucket-name**

**S3 Service Manual Operations**

Certainly! To create a bucket and upload a file to AWS S3 using the AWS Management Console (browser-based), follow these manual steps:

**Step 1: Log in to the AWS Management Console**

* Open your web browser and go to the [AWS Management Console](https://aws.amazon.com/console/).
* Log in to your AWS account using your credentials.

**Step 2: Access S3 Service**

* Once logged in, you'll be in the AWS Dashboard. In the "Find services" search bar, type "S3" and select "S3" from the results.

**Step 3: Create a Bucket and Configure Settings**

* In the Amazon S3 dashboard, click the "Create bucket" button.
* Fill out the bucket configuration details:
  + **Bucket name**: Enter a globally unique name for your bucket (bucket names must be unique across all of AWS).
  + **Region**: Choose the AWS region where you want to create your bucket.
* Click "Create."

**Step 4: Make the Bucket Public**

* After creating the bucket, click on the newly created bucket's name to open it.
* In the bucket properties, click on the "Permissions" tab.
* Under "Block public access (bucket settings)," click the "Edit" button.
* Uncheck all options to allow public access if you want the bucket to be public. Note that this makes all objects in the bucket public by default.
* Click "Save changes."

**Step 5: Enable Versioning**

* Still in the bucket properties, click on the "Management" tab.
* Click the "Enable versioning" button.
* A confirmation dialog will appear. Click "Enable versioning" to enable versioning for the bucket.

Now, your bucket is configured to be public, and versioning is enabled. Please ensure you understand the implications of making a bucket public, as it will make all objects in the bucket accessible to the public unless you further restrict access using bucket policies.

**S3 Python code to upload download**

Certainly! To create a bucket and upload a file to AWS S3 using the AWS Management Console (browser-

**Download:**

import boto3

def get\_csv\_to\_s3(bucket\_name, s3\_key, file\_name):

s3 = boto3.client('s3')

try:

s3.download\_file(bucket\_name, s3\_key, file\_name)

print("File download successfully to S3")

except Exception as e:

print(f"An error occurred: {str(e)}")

bucket\_name = 'etls3bucket11'

file\_name = 'data.csv'

s3\_key = 'raw.csv'

get\_csv\_to\_s3(bucket\_name, s3\_key, file\_name)

**Upload:**

import boto3

def upload\_csv\_to\_s3(bucket\_name, file\_name, s3\_key):

s3 = boto3.client('s3')

try:

s3 commands3.upload\_file(file\_name, bucket\_name, s3\_key)

s3 commandprint("File uploaded successfully to S3")

except FileNotFoundError:

s3 commandprint("The file was not found")

except Exception as e:

s3 commandprint(f"An error occurred: {str(e)}")

bucket\_name = 'etls3bucket11'

file\_name = 'data.csv'

s3\_key = 'new/data.csv'

upload\_csv\_to\_s3(bucket\_name, file\_name, s3\_key)