Setup:

To use Amazon S3, it is required to create an AWS Account and verify the details by updating the credit/debit card details on it. For a year Amazon offers free usage of its services. However, for verification and security, it is required to enter the payment details. After a year, the charges are leveraged on pay per use basis.

In order to use Amazon S3 using AWS Command Line Interface (CLI), the first step is to download AWS CLI and configure it.

The steps to download AWS CLI are:

1. Go to the link [AWS Command Line Interface (amazon.com)](https://aws.amazon.com/cli/).
2. Download and run the 64-bit Windows installer, for Windows operating system or MacOS PKG installer, for Mac operating system. Download, unzip, and then run the Linux installer for the Linux operating system.
3. For Mac OS, open the .pkg file. “Install AWS Command Line Interface” opens, then click on Continue. Enter the password and installation is a success. Similar is the process for other operating systems.
4. Open terminal and run the command: aws --version. If the output of the command is of the form “*aws-cli/2.1.33 Python/3.8.8 Darwin/20.3.0 exe/x86\_64 prompt/off*”, then this confirms that AWS CLI is successfully installed on the machine and its version is 2.

Now, to start with the configuration process, it is required to download the access keys from AWS Console.

The steps to download the access keys are:

1. Sign in to AWS Console. From the drop down next to the username go to “My Security Credentials”.
2. Click on “Access keys (access key ID and secret access key)”. If there is already an Access ID created and its password is known, then that can be used, otherwise, it is required to create a new access key.
3. Click on “Create New Access Keys”. View the Access Key ID and Secret Access Key and save or download these.

It is important to note here that the Secret Access Key is visible only once and once the dialog box is closed the secret access keys cannot be viewed again. It should be saved or downloaded at this point itself. If the access keys are lost, then they can be deactivated and deleted and news ones can be created.

The steps to configure AWS CLI are:

1. Open the terminal and run the command: aws configure
2. The following values need to be entered one by one:

* “AWS Access Key ID [None]” (Access Key ID)
* “AWS Secret Access Key [None]” (Secret Access Key)
* “Default region name [None]” (Region code e.g: us-east-1)
* “Default output format [None]” (Desired format of the output - json, yaml, yaml-stream, text format)

After this the configuration is done and AWS CLI commands can be run successfully. A .aws folder is created on the machine after the configuration is complete. This folder consists of two files, “config” and “credentials”. The CLI automatically logs in with the default credentials using these two files and performs the actions of the commands that are run. A user can also create his or her own profile too.

AWS S3 CLI Commands:

1. aws s3 ls

The output of this command lists all the existing buckets in the particular region.

1. aws s3 ls s3://<bucket\_name>

The output of this command enlists all the objects in the specified s3 bucket.

1. aws s3 ls s3://<bucket\_name>/<folder\_name>

The output of this command enlists all the files in the specified folder of the corresponding bucket.

1. aws s3 mb s3://<bucket\_name>

aws s3 mb s3://<bucket\_name> --region <region>

This command creates a new bucket of the specified name. If the region code is not included, then it takes into account the default region.

1. aws s3 cp <local\_filename> s3://<bucket\_name>

This command is used to upload the local file in the specified bucket. The file needs to be present in the current active directory in which the command is run. The storage class here is Standard.

1. aws s3 cp <local\_filename> s3://<bucket\_name> --storage-class REDUCED\_REDUNDANCY

This command is used to upload the local file in the specified bucket in the specified storage class, REDUCED\_REDUNDANCY. The file needs to be present in the current active directory in which the command is run. This is done when the file is less important and is not to be used frequently.

1. aws s3 sync . s3://<bucket\_name>

This command is used to sync the contents of the local folder with the specified bucket. The folder needs to be present in the current active directory in which the command is run.

1. aws s3 sync . s3://<bucket\_name> --delete

This command syncs the local folder with the S3 bucket with deletion. If any files are deleted from the local folder, the changes are reflected in the bucket as well after the execution of this command.

1. aws s3 sync s3://<bucket\_name> .

This command is used to sync the specified S3 bucket with the local folder or the current local folder. If any files are updated to the S3 bucket from any other source, then they will be reflected to the local folder as well.

1. aws s3 sync s3://<bucket\_name> . --delete

This command syncs the S3 bucket to the local folder with deletion. If any files are deleted from the bucket, the changes are reflected in the current local folder as well after the execution of this command.

1. aws s3 sync . s3://<bucket\_name> --exclude \*.css

This command syncs the local folder with the S3 bucket, however, it does not upload the files with .css extension. The specified extension files are excluded from the uploading process.

1. aws s3 sync . s3://<bucket\_name> --include \*.css

This command syncs the local folder with the S3 bucket including the files with the specified extension.

1. aws s3 mv s3://<bucket\_name> . --recursive

This command moves all the S3 bucket contents to the local active folder, thus making the bucket empty. The contents are moved and hence are deleted from the bucket. To ensure that the subdirectories are also moved, --recursive is also included in the command.

1. aws s3 mv . s3://<bucket\_name> --recursive

This command moves all the contents of the current local folder to the S3 bucket, thus making the local folder empty. The contents are moved and hence are deleted from the local folder. To ensure that the subdirectories are also moved, --recursive is also included in the command.

1. aws s3 rb s3://<bucket\_name>

This command deletes the specified S3 bucket. However, this command works only when the bucket to be deleted is empty.

1. aws s3 rb s3://<bucket\_name> --force

This command is executed when a S3 bucket has contents, i.e., files and folders and this bucket is required to be deleted. This command deletes the bucket along with contents inside it.