Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere.

S3 storage provides the following key features:

* **Buckets**—data is stored in buckets. Each bucket can store an unlimited amount of unstructured data.
* **Elastic scalability**—S3 has no storage limit. Individual objects can be up to 5TB in size.
* **Flexible data structure**—each object is identified using a unique key, and you can use metadata to flexibly organize data.
* **Downloading data—**easily share data with anyone inside or outside your organization and enable them to download data over the Internet.
* **Permissions—**assign permissions at the bucket or object level to ensure only authorized users can access data.

Boto3 is the name of the Python SDK/Library/Module/API for AWS.

 Boto3 allows us to directly create, update, and delete AWS services from our Python scripts.

The AWS Command Line Interface (AWS CLI) is a unified tool to manage

your AWS services. With just one tool to download and configure, you can

control multiple AWS services from the command line and automate them

through scripts.

**Creating an S3 Bucket**

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

1. Navigate to the [AWS Management Console](https://aws.amazon.com/console/).
2. Sign in with your AWS account credentials.

**Step 2: Access S3 Service**

1. In the AWS Management Console, search for **S3** in the search bar.
2. Click on **S3** to open the service dashboard.

**Step 3: Create a New Bucket**

1. Click on the **Create bucket** button.
2. Enter a unique bucket name (e.g., my-unique-bucket-name).
3. Select the AWS Region where you want to create the bucket.
4. Configure options such as versioning and logging as needed.
5. Set permissions to control access to the bucket.
6. Click **Create bucket**.

**Uploading Objects to S3**

**Step 1: Open Your Bucket**

1. Click on the bucket you just created.

**Step 2: Upload Objects**

1. Click the **Upload** button.
2. Drag and drop files or click on **Add files** to select files from your computer.
3. Click **Upload**.

To connect to Amazon S3 using Python on a Windows system, you'll need the **Boto3** library,AWS CLI.

**IAM User**: Create an IAM user with S3 access and obtain the Access Key ID and Secret Access Key.

## Step 1: Install python

Step 2: **Install Boto3**: You can install Boto3 using pip. Open Command Prompt and run:

pip install boto3

Step3: **IAM User**: Create an IAM user with S3 access and obtain the Access Key ID and Secret Access Key.

## Step 4: Install AWS CLI

1. Download the AWS CLI installer from the AWS official website.
2. Run the installer and follow the prompts to complete the installation.

## Step 5: Configure AWS CLI

1. Open the Command Prompt or PowerShell.
2. Run the following command to configure the AWS CLI:

aws configure

1. Enter the following details when prompted:
   * **AWS Access Key ID**: Your access key.
   * **AWS Secret Access Key**: Your secret key.
   * **Default region name**: Your preferred AWS region (e.g., us-east-1).
   * **Default output format**: Set to json, yaml, or text (e.g., json).

## Connecting to S3 Using Python

import boto3

# Initialize a session using your default profile

session = boto3.Session()

s3 = session.resource('s3')

# List buckets

print("Listing S3 Buckets:")

for bucket in s3.buckets.all():

print(bucket.name)

# Create a new bucket

bucket\_name = 'your-unique-bucket-name'

s3.create\_bucket(Bucket=bucket\_name)

print(f"Bucket {bucket\_name} created.")

# Upload a file

s3.Bucket(bucket\_name).upload\_file('C:\\path\\to\\your\\file.txt', 'file.txt')

print("File uploaded.")

# Download the file

s3.Bucket(bucket\_name).download\_file('file.txt', 'C:\\path\\to\\download\\file.txt')

print("File downloaded.")

# Delete the file

s3.Object(bucket\_name, 'file.txt').delete()

print("File deleted.")

# Remove the bucket (ensure it's empty first)

s3.Bucket(bucket\_name).delete()

print(f"Bucket {bucket\_name} deleted.")

### Running the Script

1. Save the script to a file, e.g., s3\_example.py.
2. Open Command Prompt, navigate to the script's directory, and run:

python s3\_example.py

## Conclusion

You have successfully connected to Amazon S3 using Python on Windows and performed basic operations.

Top of Form

Bottom of Form