

AWS S3 CREATION USING PYTHON BOTO3

Pre-requisites

1. Python setup in your pc or laptop
2. Boto3 library installation
3. IAM user with Access Keys

Module

- Boto3

1.Bucket Creation

The screenshot displays a dual-pane view. The left pane shows a Python script in a code editor, and the right pane shows the AWS Management Console.

Python Script (Left Pane):

```
1 import boto3
2
3 #creating an object for s3 service
4
5 s3_client=boto3.client("s3",
6     aws_access_key_id="AKIAKCPHMCSTJ1Z7EKS",
7     aws_secret_access_key="KZtTEACaMYZfK0oIhgDeLASH+LUSPXEqj+BsWml",
8 )
9
10 #the above code will connect you to aws3 run the code if you get error then we are not connected
11 # if no error you are connected to s3
12
13
14 #S3 bucket creation using boto3
15 response = s3_client.create_bucket(
16     Bucket='awsboto3-bucket-bykiran'
17 )
18 print(response)
```

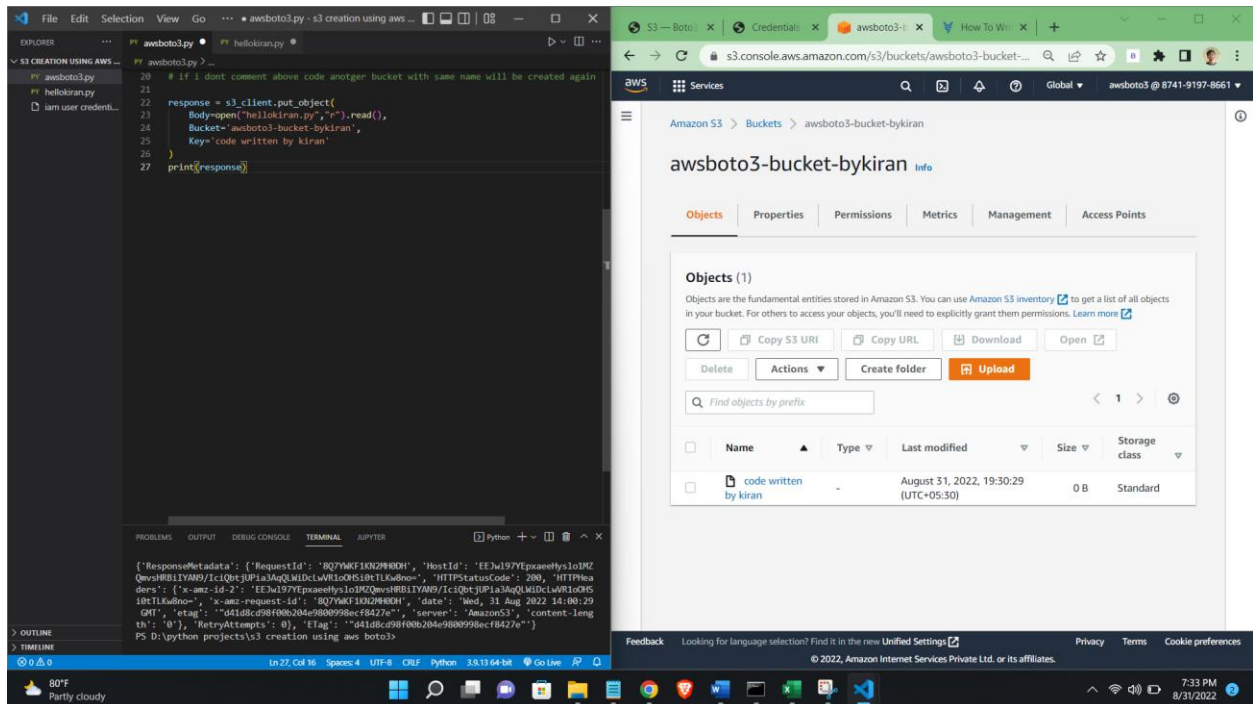
AWS Management Console (Right Pane):

The console shows the 'Buckets (5) info' page. A table lists the created buckets. The bucket 'awsboto3-bucket-bykiran' is highlighted with a red circle.

Name	AWS Region	Access	Creation date
awsboto3-bucket-bykiran	US East (N. Virginia) us-east-1	Objects can be public	August 31, 2022, 18:58:58 (UTC+05:30)
bucket54726	US East (N. Virginia) us-east-1	Bucket and objects not public	May 30, 2022, 09:11:15 (UTC+05:30)
kiran1stbucket	Asia Pacific (Mumbai) ap-south-1	Bucket and objects not public	May 3, 2022, 14:22:27 (UTC+05:30)
s3frompythonboto3	Asia Pacific (Mumbai) ap-south-1	Objects can be public	June 17, 2022, 16:13:13 (UTC+05:30)
s3fromvisualstudiopythonboto3	Asia Pacific (Mumbai) ap-south-1	Objects can be public	June 17, 2022, 17:56:48 (UTC+05:30)

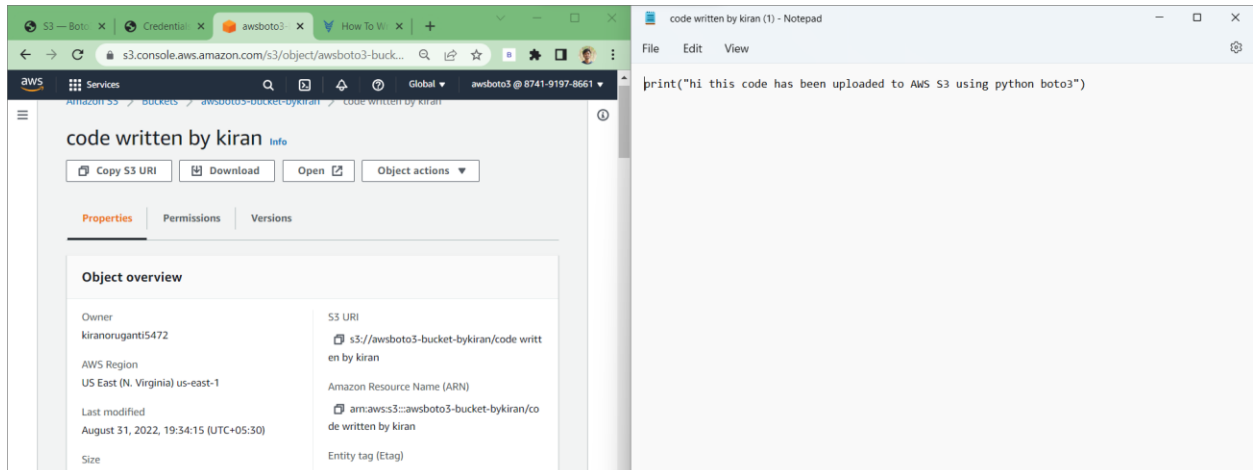
Bucket named “awsboto3-bucket-bykiran” has been created

2.Uploading file into AWS s3 using boto3



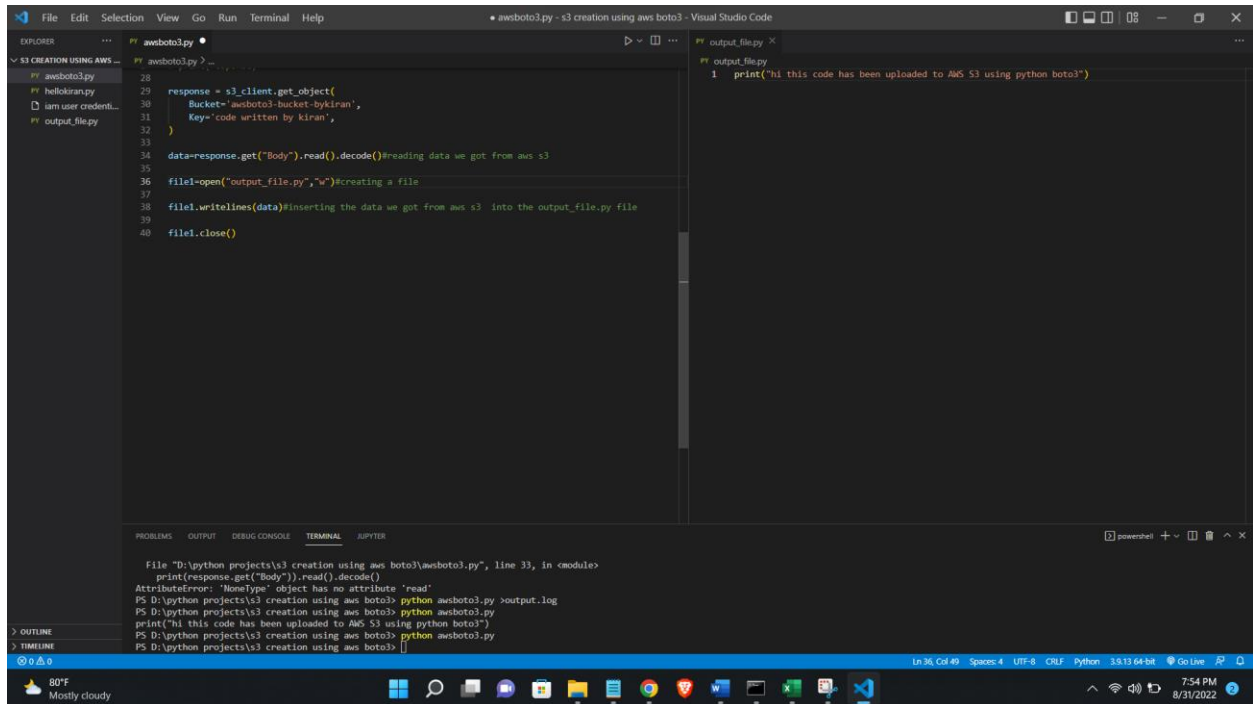
File has been uploaded. We have uploaded `hellokiran.py` file into our bucket

If you click on file and download or open it you can see the code you have written in `hellokiran.py` file code in it



Like this we can upload number of files changing the body and key

3.Download a file to local system using python boto3



```
File Edit Selection View Go Run Terminal Help
• awsboto3.py - s3 creation using aws boto3 - Visual Studio Code

EXPLORER
  s3 creation using aws boto3
  awsboto3.py
  hellokiran.py
  iam user credentials.py
  output_file.py

awsboto3.py
28
29 response = s3_client.get_object(
30     Bucket='awsboto3-bucket-bykiran',
31     Key='code written by kiran',
32 )
33
34 data=response.get("Body").read().decode()#reading data we got from aws s3
35
36 file=open("output_file.py","w")#creating a file
37
38 file.write(data)#inserting the data we got from aws s3 into the output_file.py file
39
40 file.close()

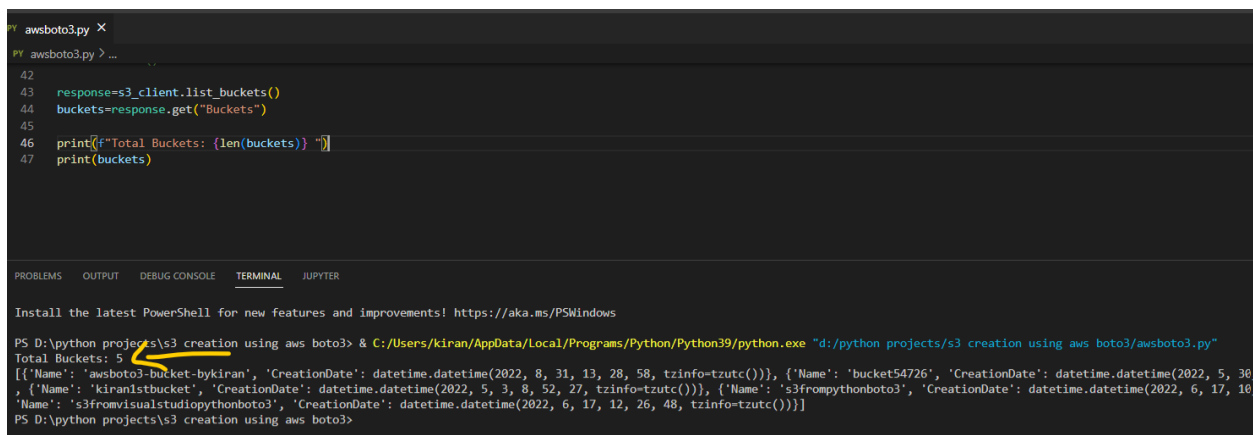
output_file.py
1 print("hi this code has been uploaded to AWS S3 using python boto3")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

File "D:\python projects\s3 creation using aws boto3\awsboto3.py", line 33, in <module>
    print(response.get("Body").read().decode()
AttributeError: 'NoneType' object has no attribute 'read'
PS D:\python projects\s3 creation using aws boto3> python awsboto3.py >output.log
PS D:\python projects\s3 creation using aws boto3> python awsboto3.py
print("hi this code has been uploaded to AWS S3 using python boto3")
PS D:\python projects\s3 creation using aws boto3> python awsboto3.py
PS D:\python projects\s3 creation using aws boto3>

80°F Mostly cloudy 7:54 PM 8/31/2022
```

4.Listing all the buckets available in AWS S3



```
awsboto3.py
awsboto3.py
42
43 response=s3_client.list_buckets()
44 buckets=response.get("Buckets")
45
46 print(f"Total Buckets: {len(buckets)}")
47 print(buckets)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\python projects\s3 creation using aws boto3> C:\Users\kiran\AppData\Local\Programs\Python\Python39\python.exe "d:\python projects\s3 creation using aws boto3\awsboto3.py"
Total Buckets: 5
[{'Name': 'awsboto3-bucket-bykiran', 'CreationDate': datetime.datetime(2022, 8, 31, 13, 28, 58, tzinfo=tzutc())}, {'Name': 'bucket54726', 'CreationDate': datetime.datetime(2022, 5, 30, 12, 12, 12, tzinfo=tzutc())}, {'Name': 'kiran1stbucket', 'CreationDate': datetime.datetime(2022, 5, 3, 8, 52, 27, tzinfo=tzutc())}, {'Name': 's3frompythonboto3', 'CreationDate': datetime.datetime(2022, 6, 17, 10, 10, 10, tzinfo=tzutc())}, {'Name': 's3fromvisualstudiopythonboto3', 'CreationDate': datetime.datetime(2022, 6, 17, 12, 26, 48, tzinfo=tzutc())}]
PS D:\python projects\s3 creation using aws boto3>
```

we have 5 buckets and we can also see the 5 buckets name

5. Listing objects or files in AWS S3

The image displays a side-by-side comparison of a local development environment and the AWS S3 console. On the left, a Visual Studio Code editor shows a Python script named `awsboto3.py`. The script uses the `boto3` library to list objects in an S3 bucket named `awsboto3-bucket-bykiran`. The terminal output shows the script's execution, listing the objects in the bucket. A yellow arrow points from the terminal output to the AWS S3 console on the right. The console shows the bucket `awsboto3-bucket-bykiran` with one object listed: `code written by kiran`. A red arrow points from the object in the console back to the terminal output.

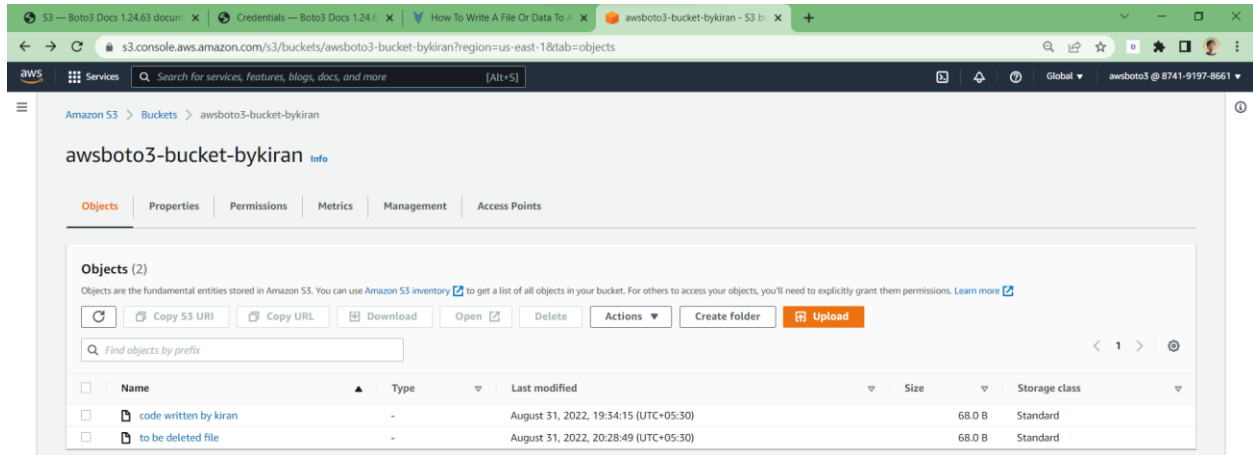
```
python/Python39/python.exe "d:\python projects\s3 creation using aws boto3\awsboto3.py"
Total Buckets: 5
[{'Name': 'awsboto3-bucket-bykiran', 'CreationDate': datetime.datetime(2022, 8, 31, 13, 28, 58, tzinfo=tzutc())}, {'Name': 'bucket54726', 'CreationDate': datetime.datetime(2022, 5, 30, 3, 41, 15, tzinfo=tzutc())}, {'Name': 'kiranlistbucket', 'CreationDate': datetime.datetime(2022, 5, 3, 8, 52, 27, tzinfo=tzutc())}, {'Name': 's3frompythonboto3', 'CreationDate': datetime.datetime(2022, 6, 17, 10, 43, 13, tzinfo=tzutc())}, {'Name': 's3fromvisualstudiopythonboto3', 'CreationDate': datetime.datetime(2022, 6, 17, 10, 43, 13, tzinfo=tzutc())}]
Total Buckets: 1
[{'Key': 'code written by kiran', 'LastModified': datetime.datetime(2022, 8, 31, 14, 4, 15, tzinfo=tzutc()), 'ETag': '"5cb762f98d8db41c8640b35807e39"', 'Size': 68, 'StorageClass': 'STANDARD', 'Owner': {'DisplayName': 'kiranoruganti15472', 'ID': 'A01bce57ebcd3890f0b7adac8321520b220b6d48d9201d3d79f80ef68c9f6'}}]
```

Name	Type	Last modified	Size	Storage class
code written by kiran	-	August 31, 2022, 19:34:15 (UTC+05:30)	68.0 B	Standard

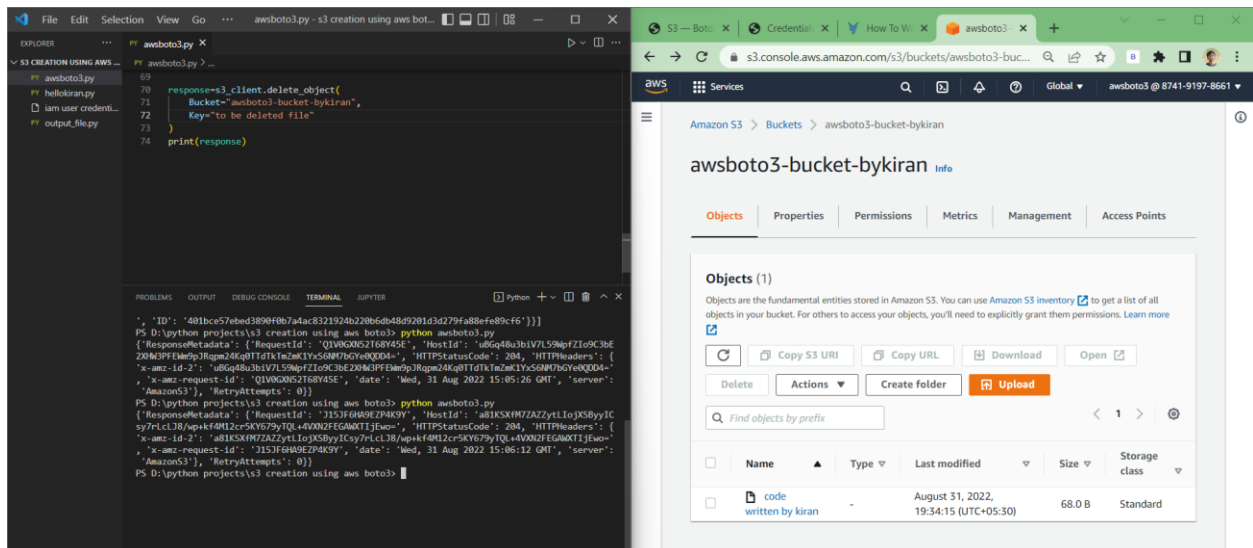
we have 1 object in our aws account and the code has shown it correctly with its name in key

6.Deleting object from AWS S3 using boto3

BEFORE

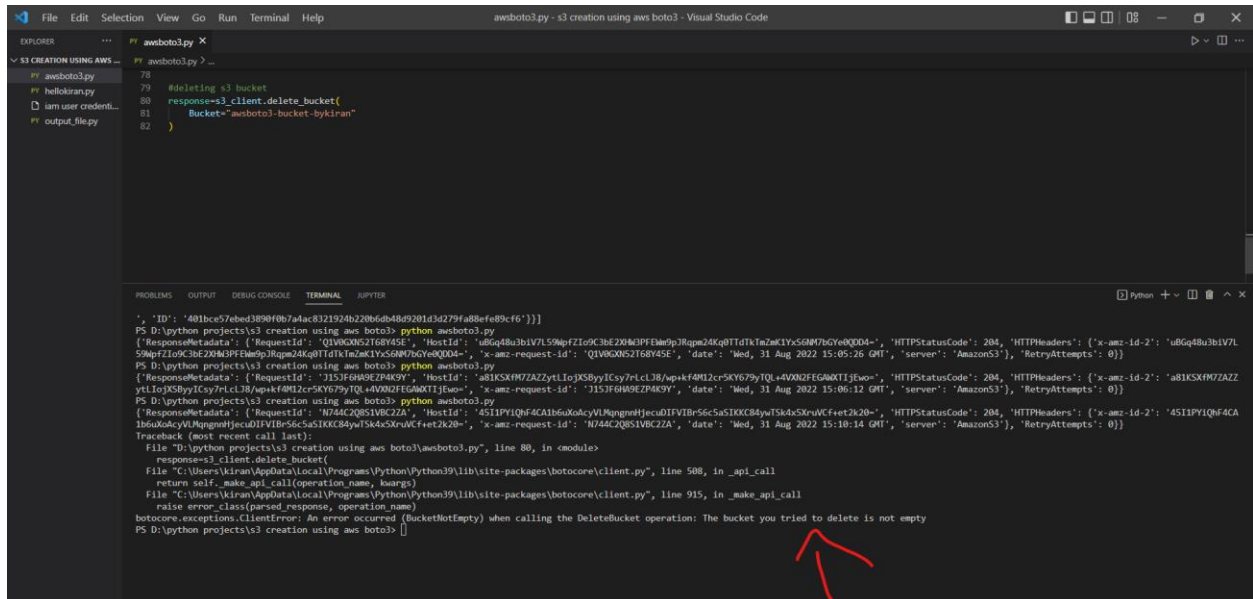


AFTER



As you can see the "to be deleted" file has been deleted

6. Deleting S3 Bucket from AWS using boto3



```
File Edit Selection View Go Run Terminal Help
awsboto3.py - s3 creation using aws boto3 - Visual Studio Code

EXPLORER
  s3 creation using aws boto3
  awsboto3.py
  helokiran.py
  iam user creden...
  output_file.py

78
79 #deleting s3 bucket
80 response=s3_client.delete_bucket(
81     Bucket="awsboto3-bucket-bykiran"
82 )

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python

", "ID": "401bce57ebd3b90f0b7d4ac8321924b22066db48d9201d3d279fad8f0e8fc6"}]]]
PS D:\python projects\s3 creation using aws boto3> python awsboto3.py
{"ResponseMetadata": {"RequestId": "Q1V8G0N52168Y4SE", "HostId": "u8Gq48u3b1V7L59mpfZ1o9C3bE2X083PFEm0pJ8qpm24kq8ITdtkTmZk1Yx56M7b6Ye0Q004+", "HTTPStatusCode": 204, "HTTPHeaders": {"x-amz-id-2": "u8Gq48u3b1V7L59mpfZ1o9C3bE2X083PFEm0pJ8qpm24kq8ITdtkTmZk1Yx56M7b6Ye0Q004+", "x-amz-request-id": "Q1V8G0N52168Y4SE", "date": "Wed, 31 Aug 2022 15:05:26 GMT", "server": "AmazonS3"}, "RetryAttempts": 0}}
PS D:\python projects\s3 creation using aws boto3> python awsboto3.py
{"ResponseMetadata": {"RequestId": "J153F6H9E2P4K9Y", "HostId": "a81KSX907AZZytlIoJXSByyICsy7rlcL38/wpkf4M12cr5KY679yTQL+4VXN2FEGWXTJjEwo-", "HTTPStatusCode": 204, "HTTPHeaders": {"x-amz-id-2": "a81KSX907AZZytlIoJXSByyICsy7rlcL38/wpkf4M12cr5KY679yTQL+4VXN2FEGWXTJjEwo-", "x-amz-request-id": "J153F6H9E2P4K9Y", "date": "Wed, 31 Aug 2022 15:06:12 GMT", "server": "AmazonS3"}, "RetryAttempts": 0}}
PS D:\python projects\s3 creation using aws boto3> python awsboto3.py
{"ResponseMetadata": {"RequestId": "N744C2Q8S1VBC2ZA", "HostId": "4511PY1QhF4CA1b6u0AcYVLqngneHJecu01FVIBr56cSa5100C84yTSk4x5XruVCFet2K20-", "HTTPStatusCode": 204, "HTTPHeaders": {"x-amz-id-2": "4511PY1QhF4CA1b6u0AcYVLqngneHJecu01FVIBr56cSa5100C84yTSk4x5XruVCFet2K20-", "x-amz-request-id": "N744C2Q8S1VBC2ZA", "date": "Wed, 31 Aug 2022 15:10:14 GMT", "server": "AmazonS3"}, "RetryAttempts": 0}}
Traceback (most recent call last):
  File "D:\python projects\s3 creation using aws boto3\awsboto3.py", line 80, in <module>
    response=s3_client.delete_bucket(
  File "C:\Users\Kiran\AppData\Local\Programs\Python\Python39\lib\site-packages\botocore\client.py", line 508, in _api_call
    return self._make_api_call(operation_name, kwargs)
  File "C:\Users\Kiran\AppData\Local\Programs\Python\Python39\lib\site-packages\botocore\client.py", line 915, in _make_api_call
    raise error_class(parsed_response, operation_name)
botocore.exceptions.ClientError: An error occurred (BucketNotEmpty) when calling the DeleteBucket operation: The bucket you tried to delete is not empty
PS D:\python projects\s3 creation using aws boto3>
```

As you can see I got an error that **Bucket we tried is not empty**. To delete the bucket we have to login to our AWS account and empty the bucket then only we can delete the bucket using programming. This is the restriction kept by AWS to ensure the users will not delete their bucket by mistakenly.