



## How can we interact with AWS in python?

we can use the boto 3, You use the AWS SDK for Python (Boto3) to create, configure, and manage AWS services, such as Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Simple Storage Service (Amazon S3). The SDK provides an object-oriented API as well as low-level access to AWS services. We can use it build automated reports, perform sentiment analysis ,etc.

## What are AWS services ?

1. IAM
2. S3
3. SNS
4. Comprehend
5. Rekognition
6. RDS
7. EC2
8. Lambda

## What is S3 ?

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.

## How can we connect to S3 Using Boto3?

### Create boto3 client

```
import boto3
s3 = boto3.client('s3', region_name='us-east-1',
                  aws_access_key_id=AWS_KEY_ID,
                  aws_secret_access_key=AWS_SECRET)
```

what is aws\_access\_key\_id, aws\_secret\_access\_key? how can we create them ?

What is bucket?



## What can we do with buckets?

### 1. How can we Create Bucket in boto3?

#### Create boto3 client

```
import boto3
s3 = boto3.client('s3', region_name='us-east-1',
                  aws_access_key_id=AWS_KEY_ID,
                  aws_secret_access_key=AWS_SECRET)
```

#### Create bucket!

```
bucket = s3.create_bucket(Bucket='gid-requests')
```

### 2. How can we list buckets?

```
import boto3
s3 = boto3.client('s3', region_name='us-east-1',
                  aws_access_key_id=AWS_KEY_ID,
                  aws_secret_access_key=AWS_SECRET)
```

#### List Buckets

```
bucket_response = s3.list_buckets()
```

### 3. How can we delete bucket?

## Deleting buckets

#### Create boto3 client

```
import boto3
s3 = boto3.client('s3', region_name='us-east-1',
                  aws_access_key_id=AWS_KEY_ID,
                  aws_secret_access_key=AWS_SECRET)
```

#### Delete Bucket

```
response = s3.delete_bucket('gid-requests')
```



## 4. What is the difference between bucket and objects??

### A Bucket



- A bucket has a **name**
- **Name** is a string
- **Unique** name in all of S3.
- Contains **many** objects

### An Object



- An object has a **key**
- **Name** is full path from bucket root
- **Unique** key in the bucket
- Can only be in **one** parent bucket

## 5. How can we upload file from boto3?

### Uploading files

```
s3.upload_file(  
    Filename='gid_requests_2019_01_01.csv',  
    Bucket='gid-requests',  
    Key='gid_requests_2019_01_01.csv')
```

## 6. How can we list objects in bucket?

### Listing objects in a bucket

```
response = s3.list_objects(  
    Bucket='gid-requests',  
    MaxKeys=2,  
    Prefix='gid_requests_2019_')  
  
print(response)
```



## 7. how can we get object from bucket using boto3?

### Getting object metadata

```
response = s3.head_object(  
    Bucket='gid-requests',  
    Key='gid_requests_2018_12_30.csv')  
  
print(response)
```

## 8. how can we download file from bucket using boto3?

### Downloading files

```
s3.download_file(  
    Filename='gid_requests_downed.csv',  
    Bucket='gid-requests',  
    Key='gid_requests_2018_12_30.csv')
```

## 9. How can we delete object from bucket using boto3?

### Deleting objects

```
s3.delete_object(  
    Bucket='gid-requests',  
    Key='gid_requests_2018_12_30.csv')
```

## 10. Can you suggest the problem here?

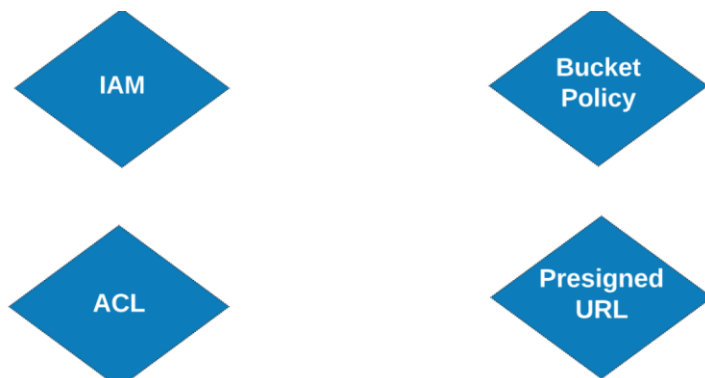
<https://gid-staging.potholes.csv> is URL to AWS object called potholes.csv  
we are trying to access an object without **permissions**

```
df = pd.read_csv('https://gid-staging.potholes.csv')
```

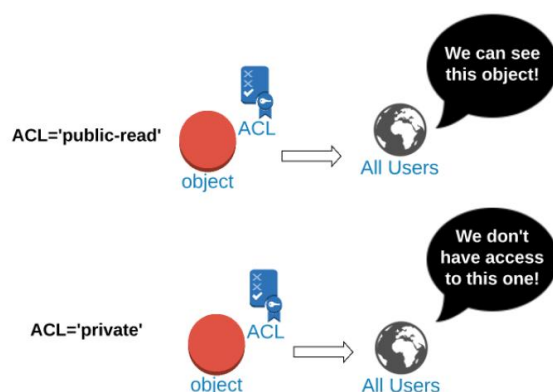
```
/usr/local/Cellar/python/3.7.1/Frameworks/Python.framework/Versions/3.7/lib/python  
647 class HTTPDefaultErrorHandler(BaseHandler):  
648     def http_error_default(self, req, fp, code, msg, hdrs):  
--> 649         raise HTTPError(req.full_url, code, msg, hdrs, fp)  
650  
651 class HTTPRedirectHandler(BaseHandler):  
  
HTTPError: HTTP Error 403: Forbidden
```



## 11. What are AWS Permissions Systems?



## 12. what are the ACLs types?



## 13. how can we put ACLS using boto3 on object ?

There are 2 ways

1.

### Setting ACLs on upload

Upload file with 'public-read' ACL

```
s3.upload_file(  
    Bucket='gid-requests',  
    Filename='potholes.csv',  
    Key='potholes.csv',  
    ExtraArgs={'ACL': 'public-read'})
```



## 2.

### Upload File

```
s3.upload_file(  
    Filename='potholes.csv', Bucket='gid-requests', Key='potholes.csv')
```

### Set ACL to 'public-read'

```
s3.put_object_acl(  
    Bucket='gid-requests', Key='potholes.csv', ACL='public-read')
```

## 14. How can we access public object?

### S3 Object URL Template

```
https://{bucket}.{key}
```

### URL for Key= '2019/potholes.csv'

```
https://gid-requests.2019/potholes.csv
```

## 15. how can we access a private object ? someone who has access can download and make it available to anyone

### Download File

```
s3.download_file(  
    Filename='potholes_local.csv',  
    Bucket='gid-staging',  
    Key='2019/potholes_private.csv')
```

### Read From Disk

```
pd.read_csv('./potholes_local.csv')
```

## 16. What is the Pre-signed URLs?

### Pre-signed URLs

- Expire after a certain timeframe
- Great for temporary access

### Example

```
https://?AWSAccessKeyId=12345&Signature=rBmnrwutb6VkJ9hE8Uub%2BBYA9mY%3D&Expires=1557624801
```



## 17. how can we create pre-signed URL to object in boto3?

### Generate Presigned URL

```
share_url = s3.generate_presigned_url(  
    ClientMethod='get_object',  
    ExpiresIn=3600,  
    Params={'Bucket': 'gid-requests', 'Key': 'potholes.csv'}  
)
```

### Open in Pandas

```
pd.read_csv(share_url)
```

## 18. how can we uploadan HTML file to S3 ?

### Upload an HTML file to S3

```
s3.upload_file(  
    Filename='./table_agg.html',  
    Bucket='datacamp-website',  
    Key='table.html',  
    ExtraArgs = {  
        'ContentType': 'text/html',  
        'ACL': 'public-read'}  
)
```