**Spring Boot with AWS S3**

**AWS S3**:

Amazon Simple Storage Service (**Amazon S3**) is an object storage service that offers industry-leading scalability, data availability, security, and performance. Customers of all sizes and industries can use Amazon S3 to store and protect any amount of data for a range of use cases, such as data lakes, websites, mobile applications, backup and restore, archive, enterprise applications, IoT devices, and big data analytics. Amazon S3 provides management features so that you can optimize, organize, and configure access to your data to meet your specific business, organizational, and compliance requirements.

**How Amazon S3 works**:

Amazon S3 is an object storage service that stores data as objects within buckets. An object is a file and any metadata that describes the file. A bucket is a container for objects.

To store your data in Amazon S3, you first create a bucket and specify a bucket name and AWS Region. Then, you upload your data to that bucket as objects in Amazon S3. Each object has a key (or key name), which is the unique identifier for the object within the bucket.

S3 provides features that you can configure to support your specific use case. For example, you can use S3 Versioning to keep multiple versions of an object in the same bucket, which allows you to restore objects that are accidentally deleted or overwritten.

Buckets and the objects in them are private and can be accessed only if you explicitly grant access permissions. You can use bucket policies, AWS Identity and Access Management (IAM) policies, access control lists (ACLs), and S3 Access Points to manage access.

**Buckets**:

A bucket is a container for objects stored in Amazon S3. You can store any number of objects in a bucket and can have up to 100 buckets in your account.

Every object is contained in a bucket. For example, if the object named photos/xyz.jpg is stored in the DOC-EXAMPLE-BUCKET bucket in the **ap-south-1** Region, then it is addressable using the URL <https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/photos/xyz.jpg>.

When you create a bucket, you enter a bucket name and choose the AWS Region where the bucket will reside. After you create a bucket, you cannot change the name of the bucket or its Region.

**Objects**:

Objects are the fundamental entities stored in Amazon S3. Objects consist of object data and metadata. The metadata is a set of name-value pairs that describe the object. These pairs include some default metadata, such as the date last modified, and standard HTTP metadata, such as Content-Type. You can also specify custom metadata at the time that the object is stored.

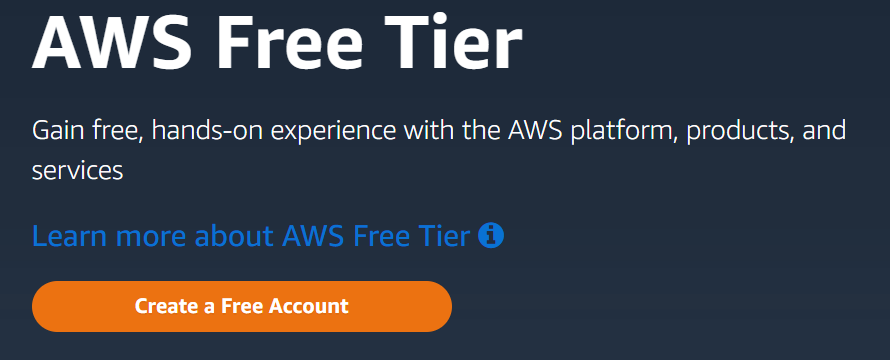
**Keys**:

An object key (or key name) is the unique identifier for an object within a bucket. Every object in a bucket has exactly one key. The combination of a bucket, object key, and optionally, version ID (if S3 Versioning is enabled for the bucket) uniquely identify each object. So, you can think of Amazon S3 as a basic data map between "bucket + key + version" and the object itself.

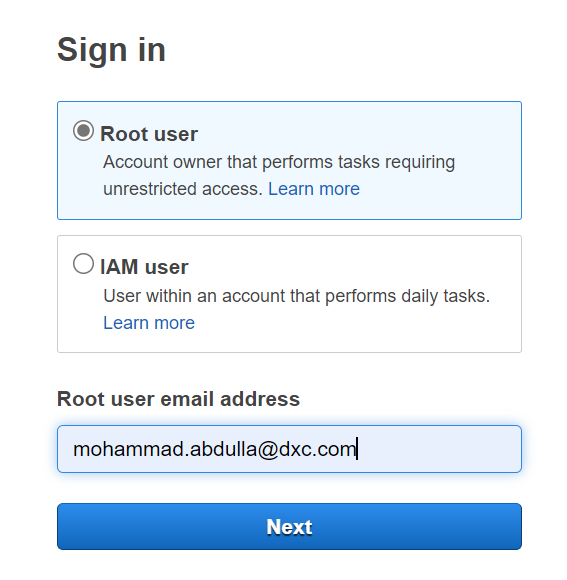
Every object in Amazon S3 can be uniquely addressed through the combination of the web service endpoint, bucket name, key, and optionally, a version. For example, in the URL https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/photos/xyz.jpg, DOC-EXAMPLE-BUCKET is the name of the bucket and photos/xyz.jpg is the key.

Create bucket and get access key:

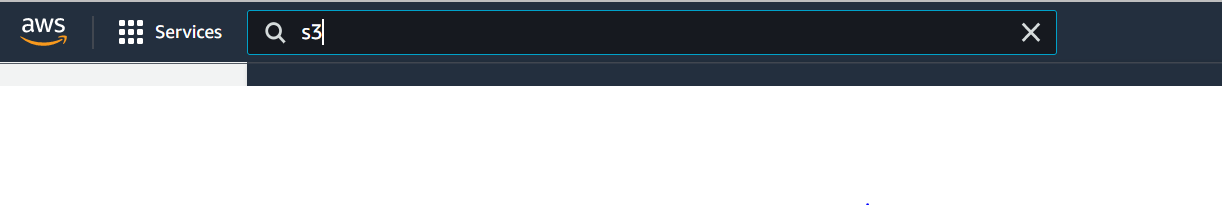
Create an account on amazon aws:

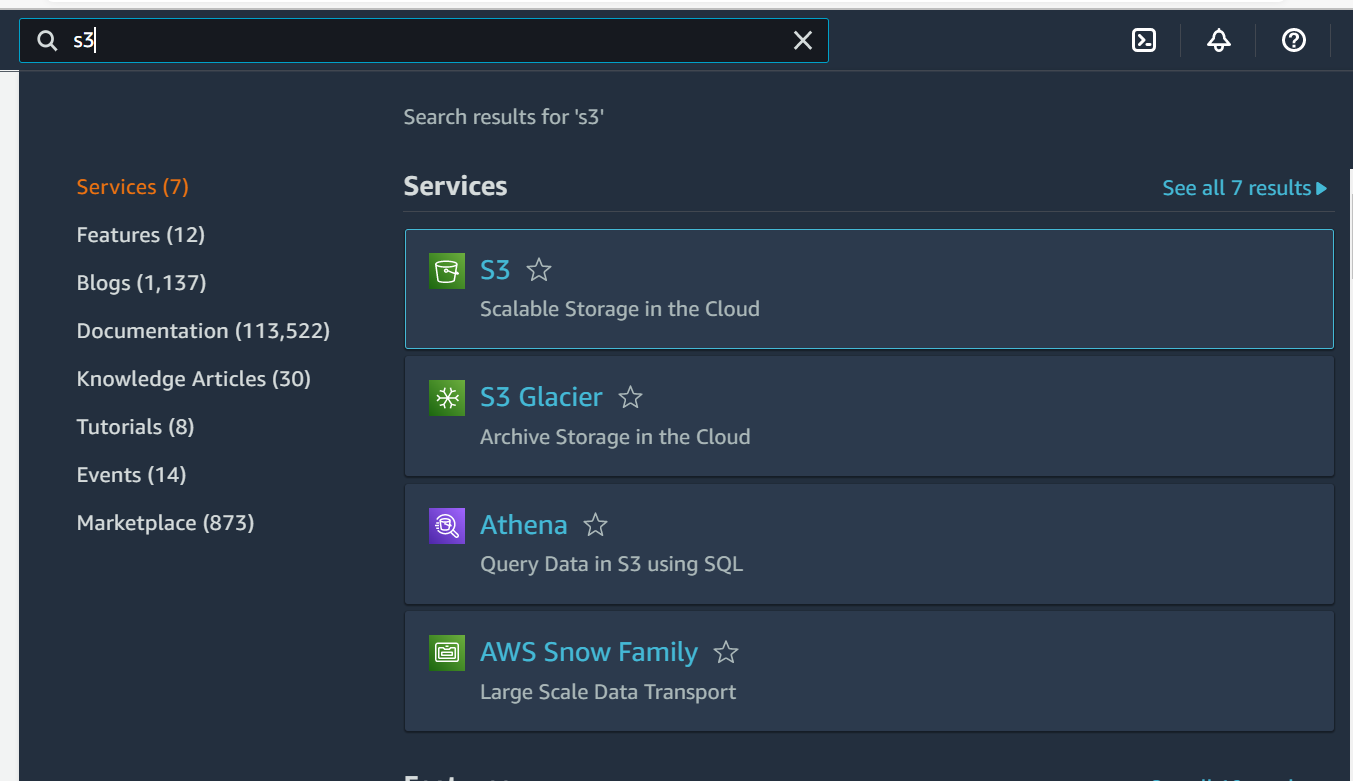


**Login to account**:

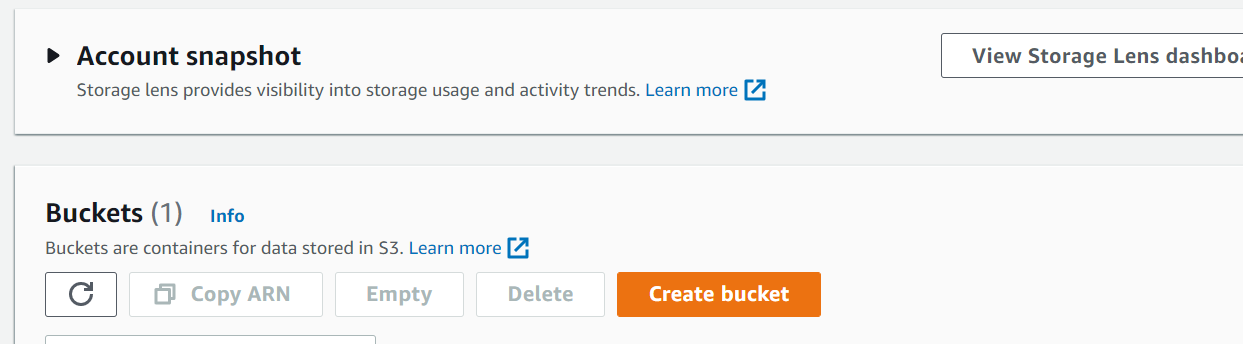


**After successful login click on search and search for S3:**

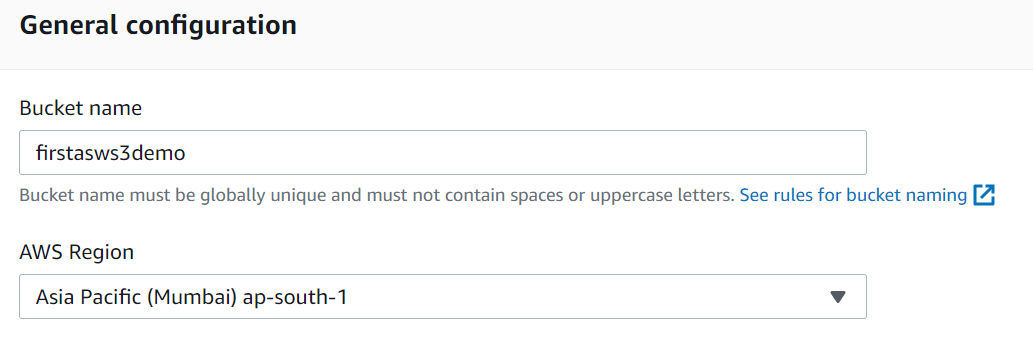




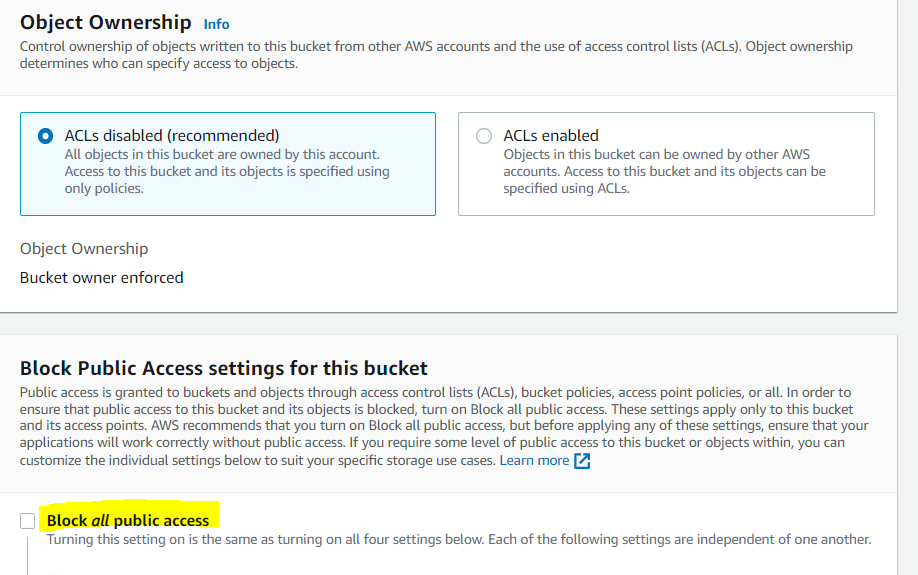
Click on S3 and then click on Create bucket button:



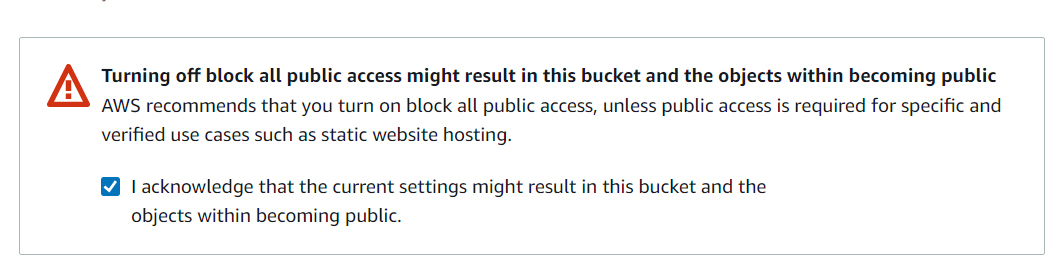
Provide the bucket name and region:



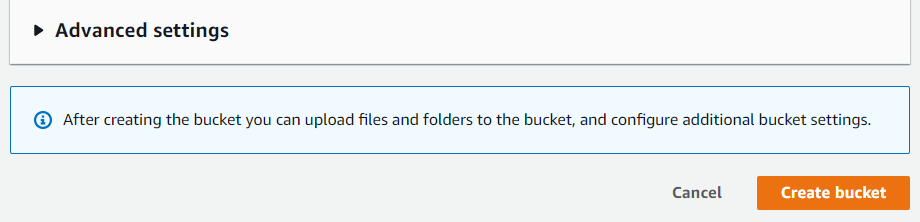
Choose the object ownership as recommended and uncheck the check box “Block all public access”.



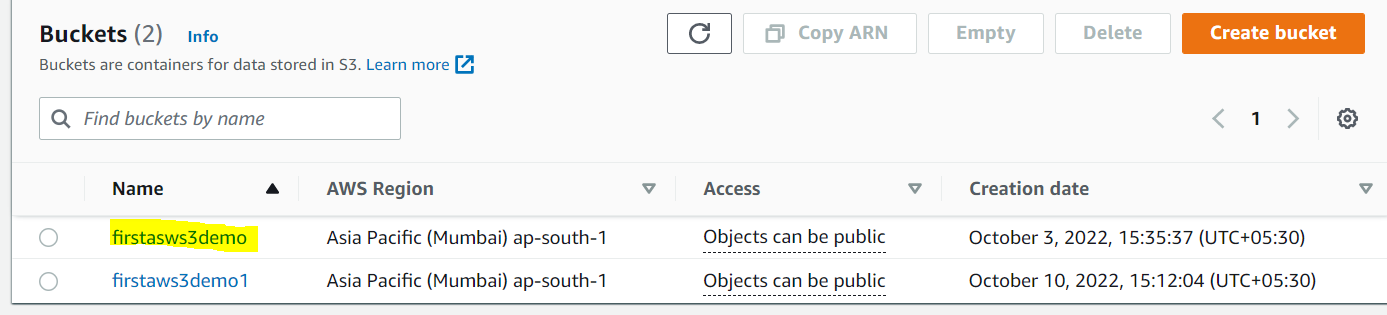
Acknowledge the public access:



Click on create bucket:

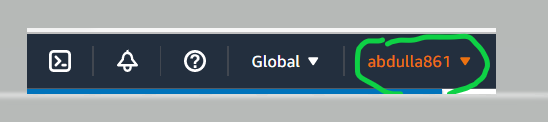


Bucket has been created:

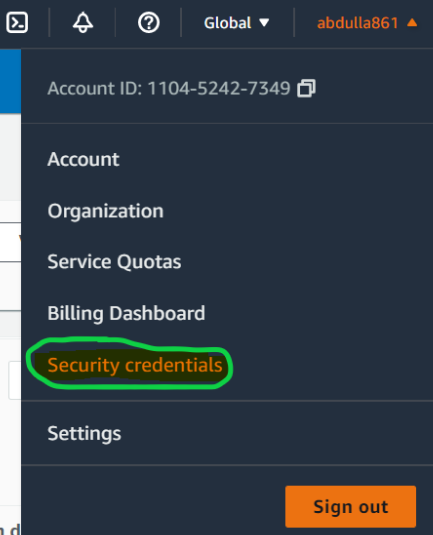


**For access key**:

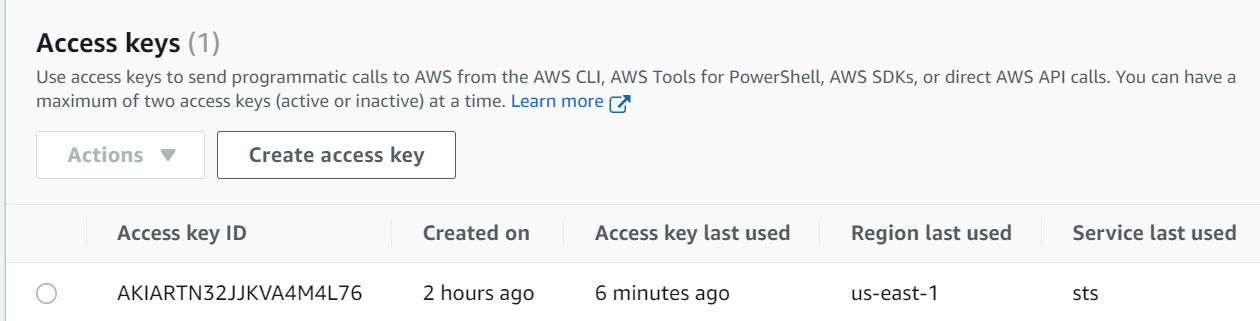
Click on user icon:



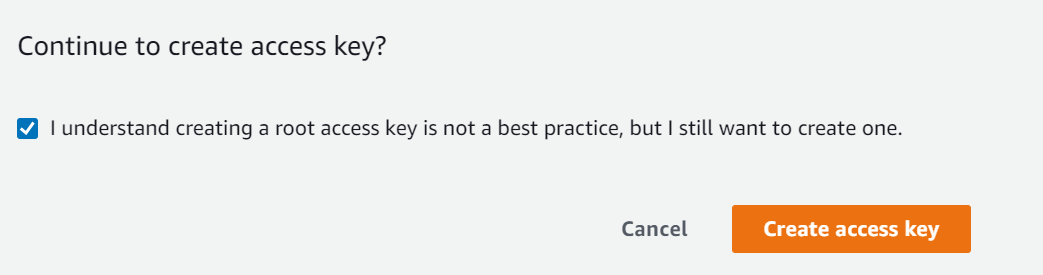
Click on security credentials:

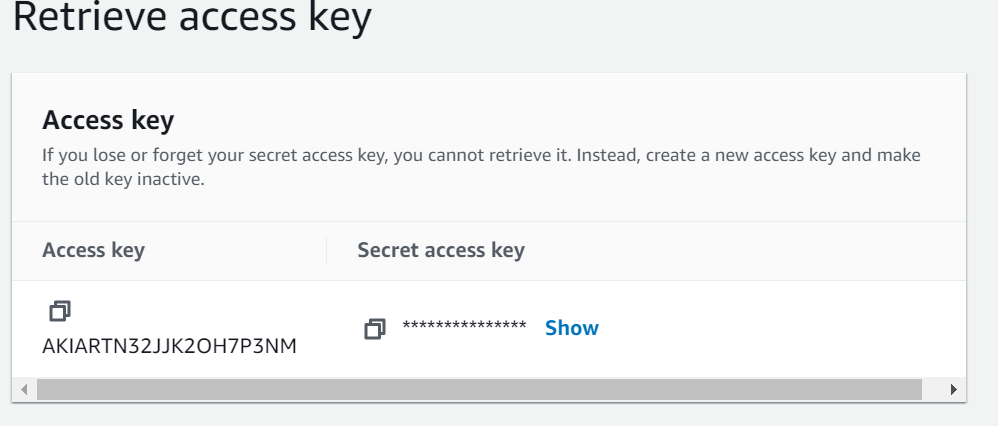


Go To Access Key and click on create access key:

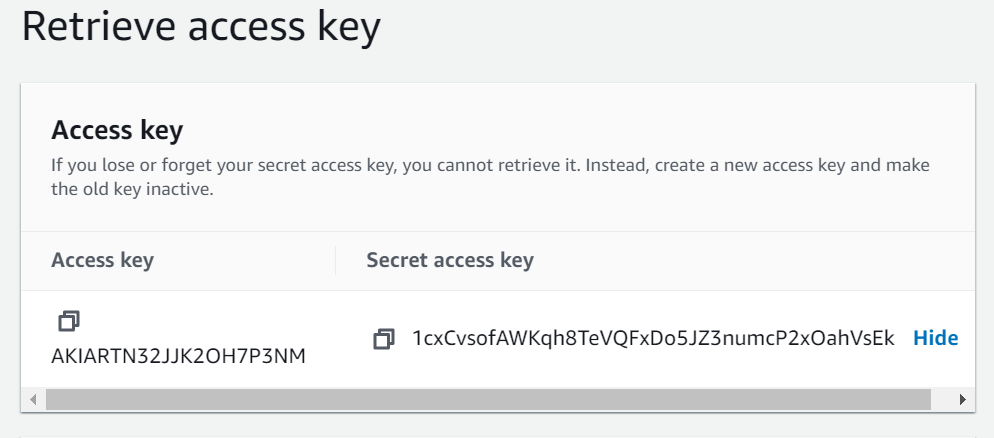


Click on button to create the access key:





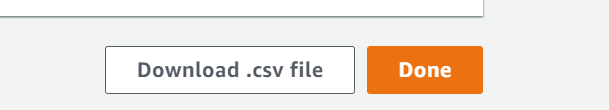
Click on show to display the secret access key:



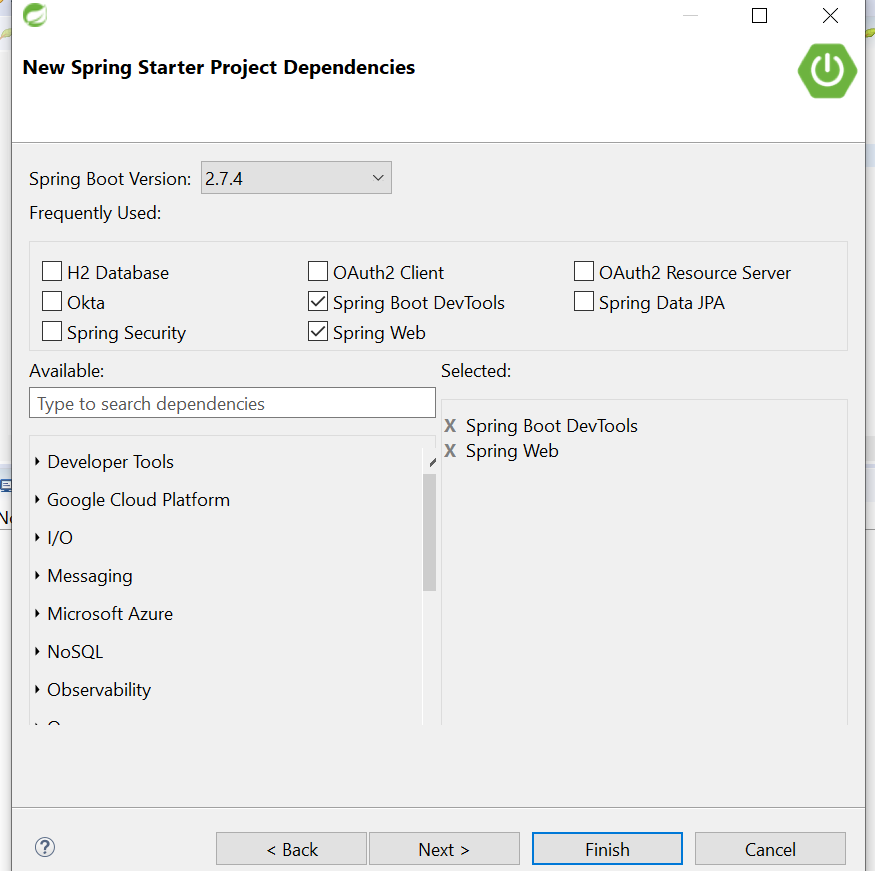
Note down the Access key and Secrete Access Key for application use.

Or

Download the .csv file:

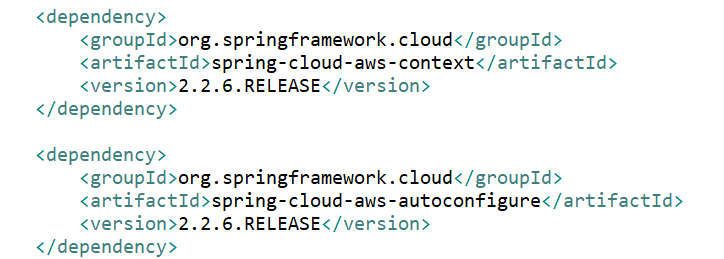


**Programming**:

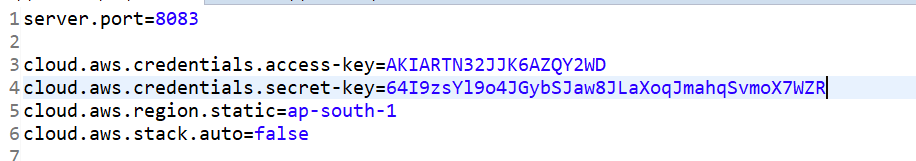
1. After creating the bucket, create a spring boot application to upload, download and delete the file:

Click finish.

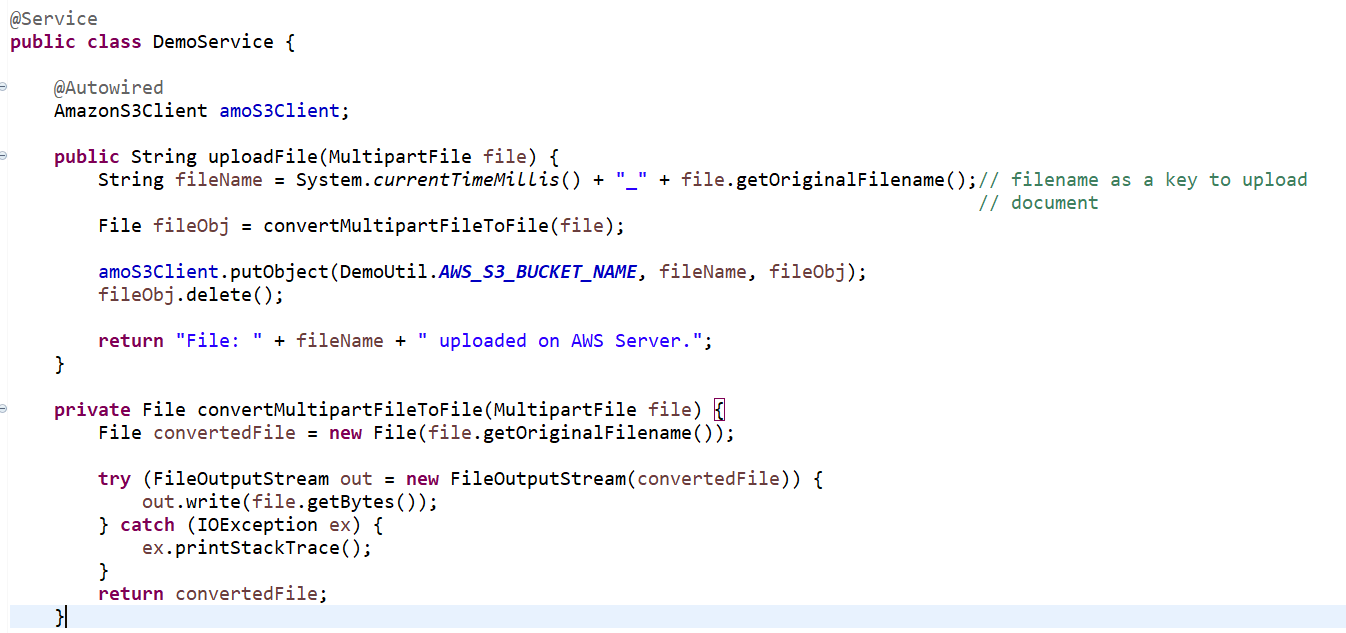
1. Add maven dependency of AWS:



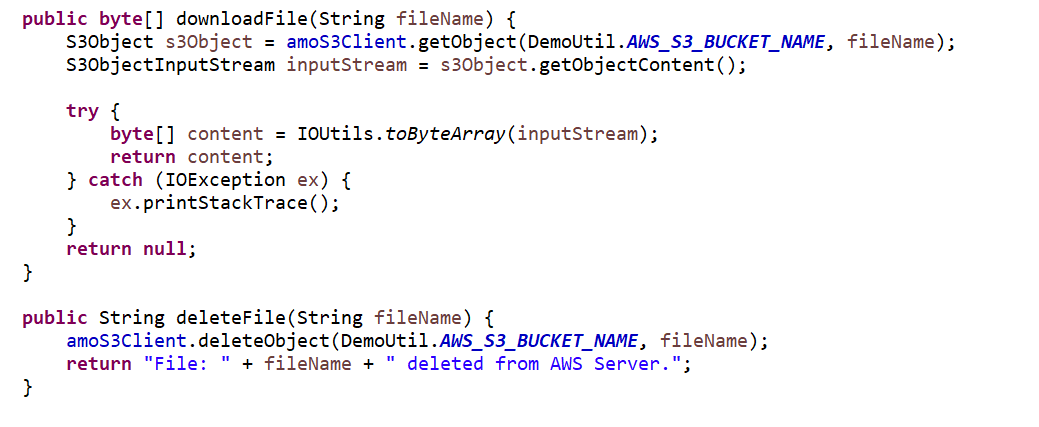
1. Add access key and secrete key into resource file (application.properties):



1. Create a service and add method to upload the file:



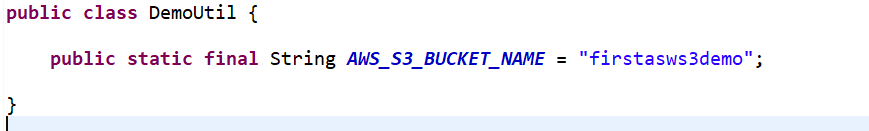
1. Add method to download and delete the file:



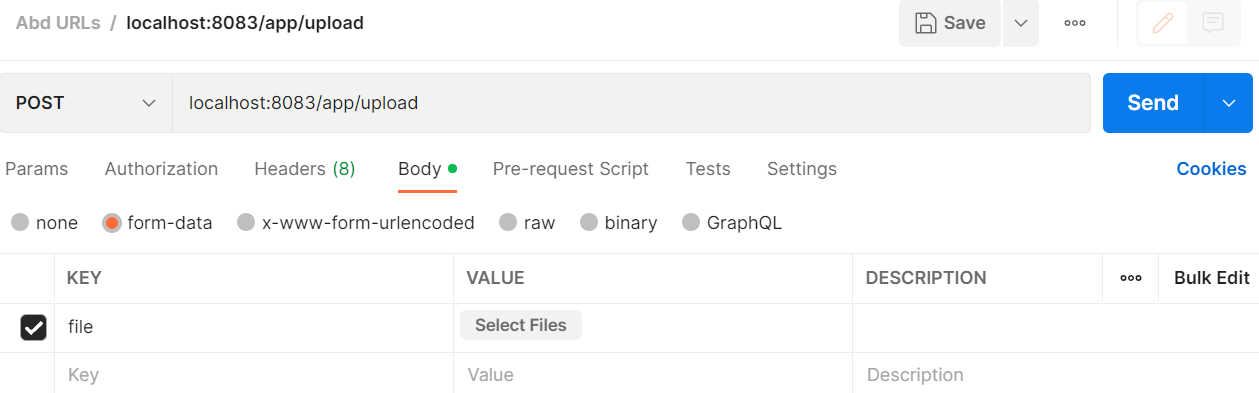
1. Create controller to generate the end to hit the services:



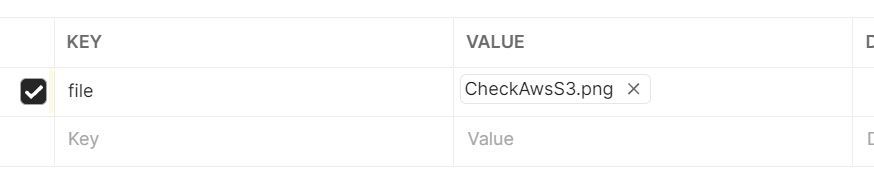
1. Create a util class to provide the bucket name:



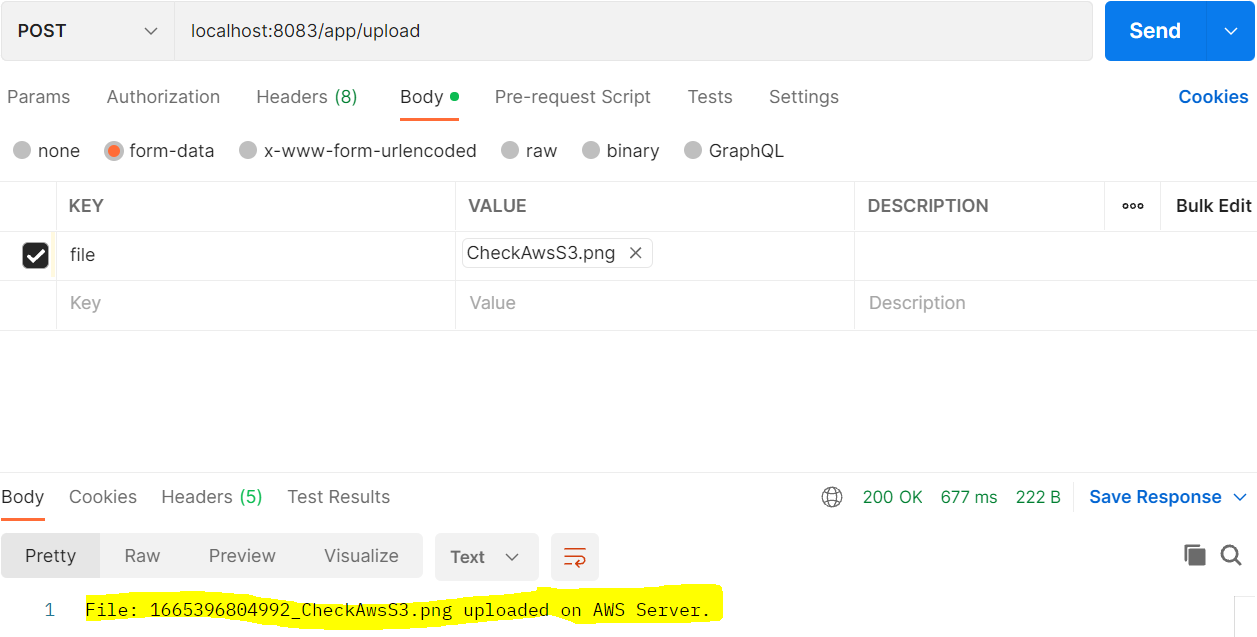
1. Now run the application and open the postman to hit the endpoints of application:
2. To upload the file:



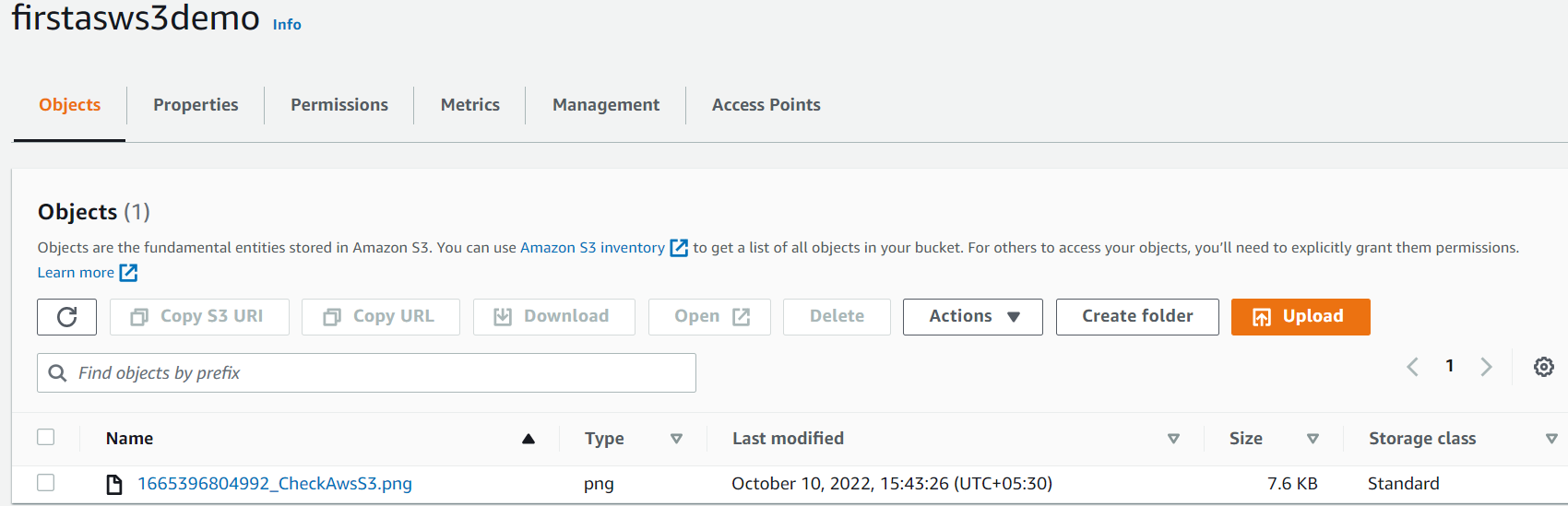
Select file to upload click send:



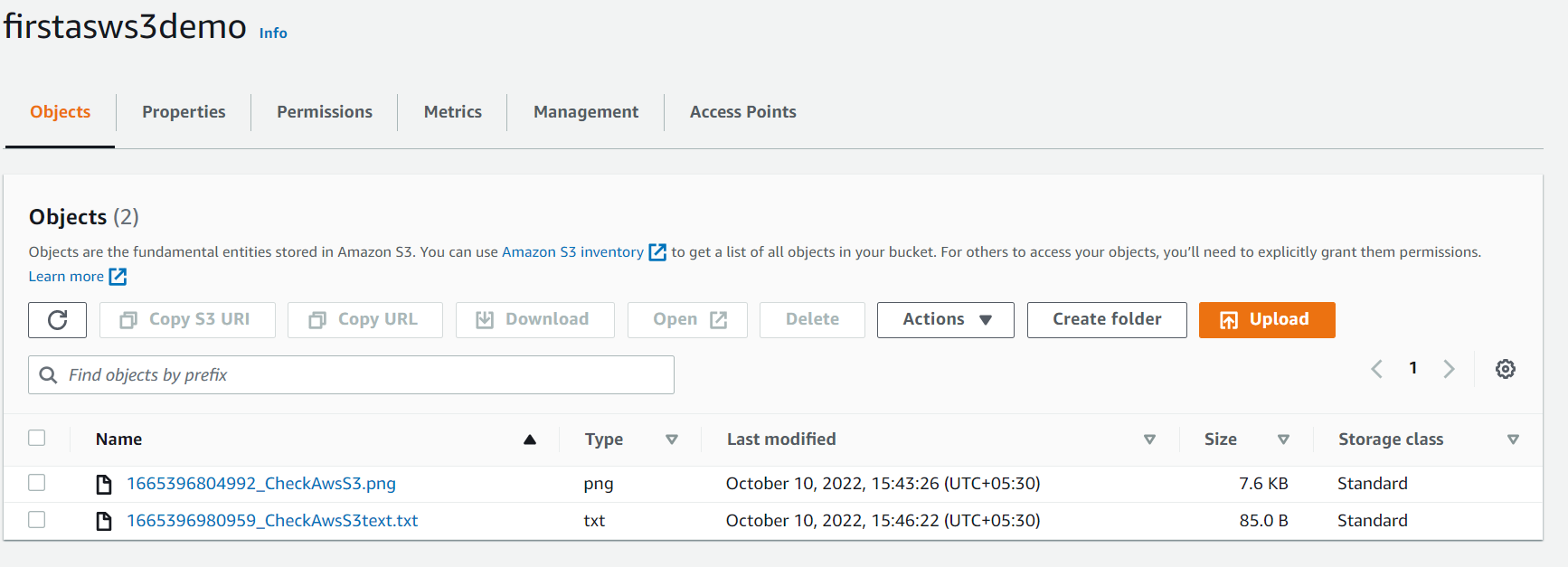
In Response you will get the response code 200 and the document uploaded successfully on AWS server:



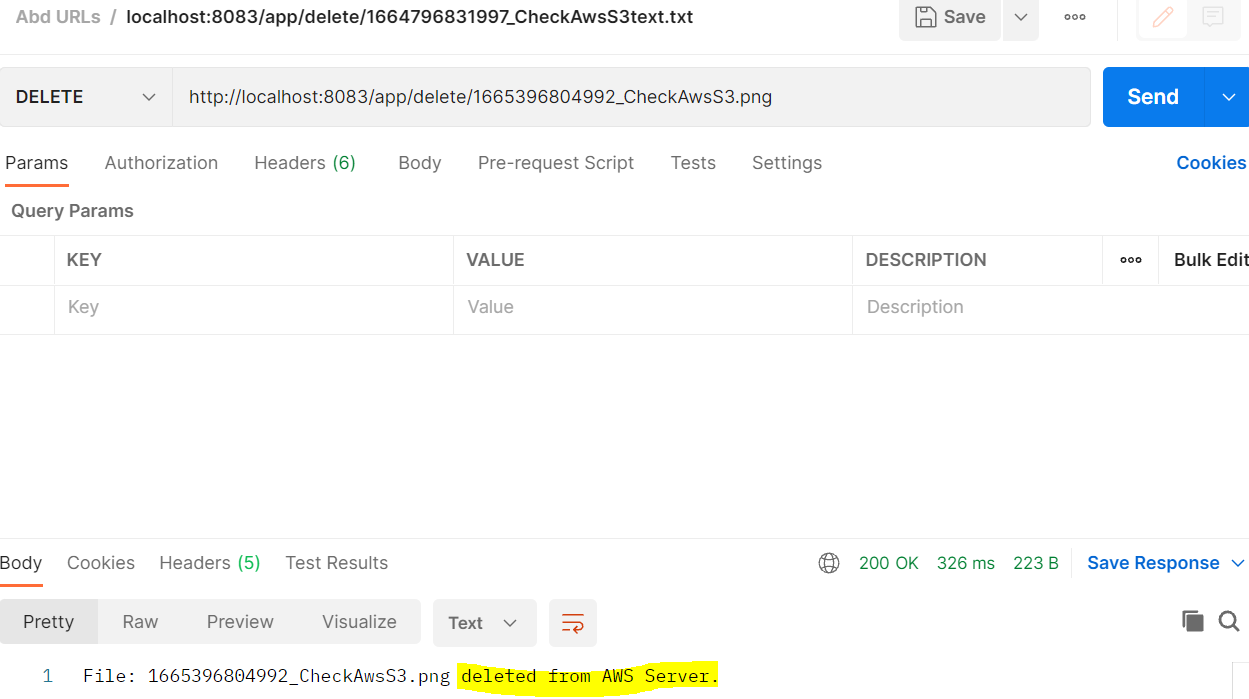
Go to amazon AWS server and check the bucket and verify the uploaded document:



Upload one more document:



1. Now perform the download URL on browser then it is downloaded the file from AWS server.
2. For delete use the postman to hit the end URL of delete:



Go t0 amazon AWS server and check the bucket and verify the document has been removed from server.

See below video for your reference:

<https://www.youtube.com/watch?v=vY7c7k8xmKE>

<https://www.youtube.com/watch?v=c3POiw8rHoQ>