DALHOUSIE UNIVERSITY

FACULTY OF COMPUTER SCIENCE

ASSIGNMENT 1: PART B

In
The Class of

CSCI5710: SERVERLESS DATA PROCESSING

by

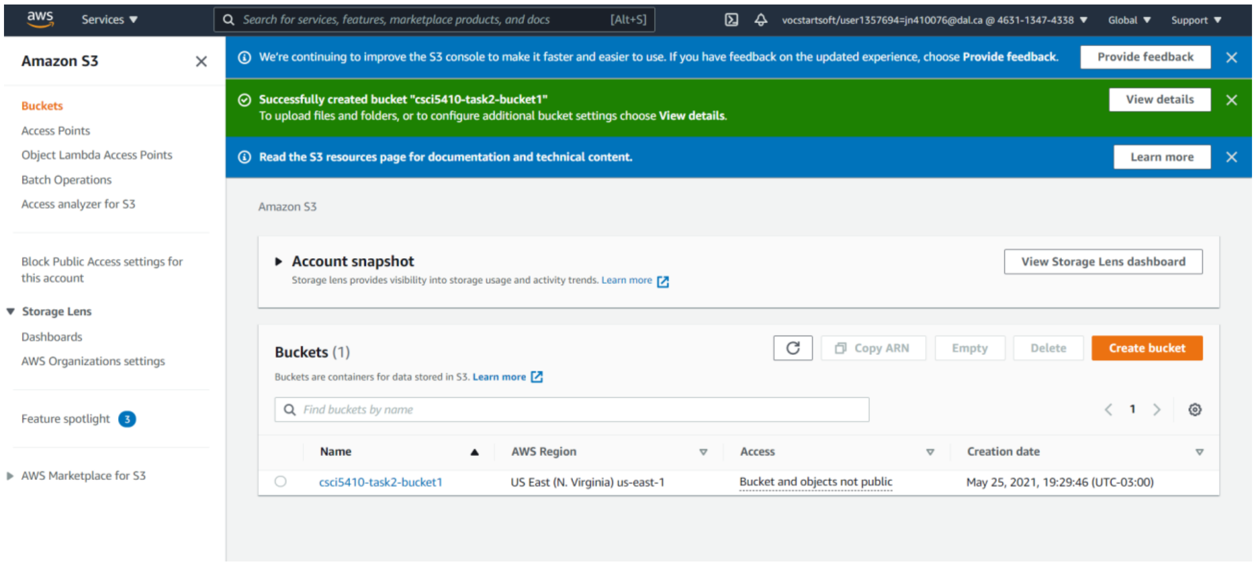
Janvi Patel [B00863421]

Submitted to
Prof. Saurabh Dey
Department of Computer Science
Dalhousie university.

Date: 25th May 2021

Part B. AWS S3 Storage experiment:

a) Create a S3 bucket from AWS management console. Once it is done, create a text file (empty file) in your computer and rename it with your “First Name”. e.g. “Alice.txt”.



We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose **Provide feedback**.

Successfully created bucket "csci5410-task2-bucket1"
To upload files and folders, or to configure additional bucket settings choose **View details**.

Read the S3 resources page for documentation and technical content.

Account snapshot
Storage lens provides visibility into storage usage and activity trends. Learn more

Buckets (1)
Buckets are containers for data stored in S3. Learn more

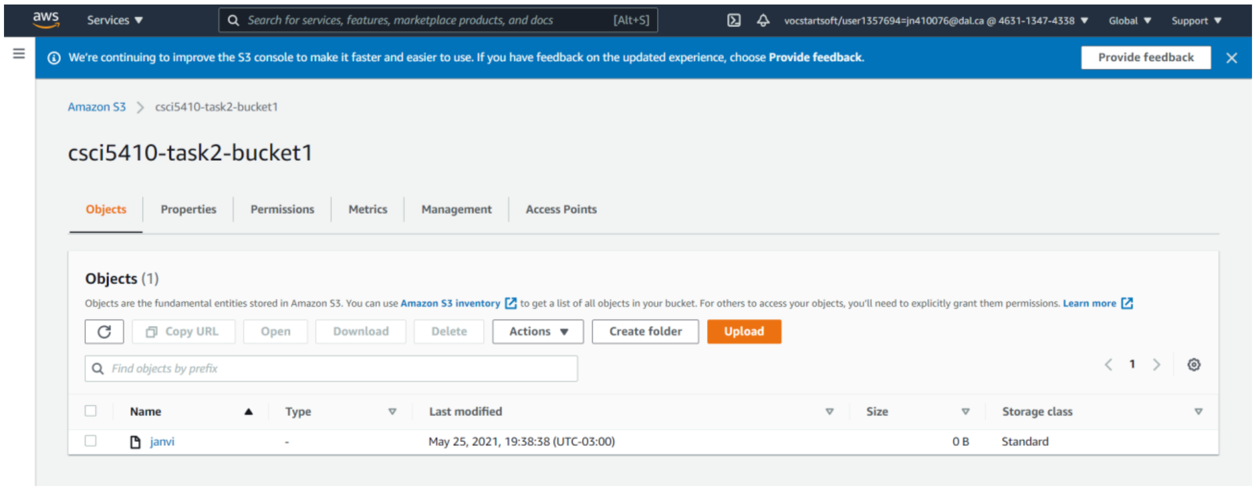
Name	AWS Region	Access	Creation date
csci5410-task2-bucket1	US East (N. Virginia) us-east-1	Bucket and objects not public	May 25, 2021, 19:29:46 (UTC-03:00)

Figure 1: Output for create bucket

b) Explore AWS SDK for Java - and using Java program written based on the SDK specification, upload the file on the S3 bucket you created.

```
package CSCI5409_Assignment1.AWSproject;

import com.amazonaws.AmazonServiceException;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import java.io.File;
import java.io.IOException;

public class UploadFile {
public static void main(String[] args) throws IOException
{
        //variable declaration for bucket name, file path and file name
        String bucketName = "csci5410-task2-bucket1";
        String filePath = "D:\\Study D\\Dalhousie\\Sem 3\\Serverless\\Assignments\\Assignment 1\\janvi.txt";
        String keyName = "janvi";
        System.out.format("Uploading %s.txt to S3 bucket %s...\n", keyName, bucketName);

        //connection with AWS s3 using region East 1 and the configurations provided in local    folder
        final AmazonS3 s3 =  AmazonS3ClientBuilder.standard().withRegion(Regions.US_EAST_1).build();

        try
        {
                //using put object uploading file present at a particular location in the specific bucket
            s3.putObject(bucketName, keyName, new File(filePath));
         }
         catch (AmazonServiceException e)              //Exception handling
```

```
                {
                    System.out.println("Error while uploading the file!");
                    System.err.println(e.getErrorMessage());
                    System.exit(1);
                 }
                System.out.println("Successfully uploaded!");
            }
    }
```



We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose **Provide feedback**.

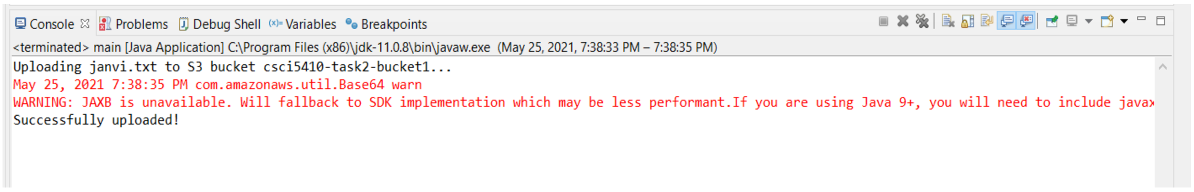
csci5410-task2-bucket1

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class
janvi	-	May 25, 2021, 19:38:38 (UTC-03:00)	0 B	Standard

Figure 2: Output for file successful upload



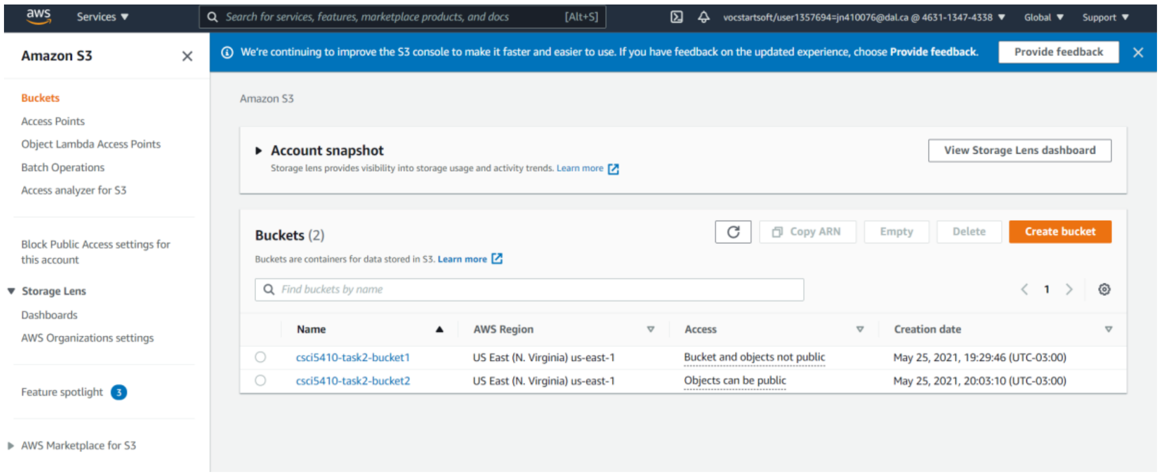
```
<terminated> main [Java Application] C:\Program Files (x86)\jdk-11.0.8\bin\javaw.exe (May 25, 2021, 7:38:33 PM – 7:38:35 PM)
Uploading janvi.txt to S3 bucket csci5410-task2-bucket1...
May 25, 2021 7:38:35 PM com.amazonaws.util.Base64 warn
WARNING: JAXB is unavailable. Will fallback to SDK implementation which may be less performant.If you are using Java 9+, you will need to include javax
Successfully uploaded!
```

Figure 3: Output for successfully uploaded

c) Create a second bucket in AWS S3 using Java, and programmatically change the access permission, "disable public access". In addition, programmatically change the ACL write option to "full-control" for bucket owner

a. Second bucket in AWS S3 using Java code

```
package CSCI5409_Assignment1.AWSproject;
import java.io.IOException;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.AmazonS3Exception;

public class CreateBucket {

public static void main(String[] args) throws IOException
{
        //connection with AWS s3 using region East 1 and the configurations provided in local folder
        final AmazonS3 s3 = AmazonS3ClientBuilder.standard().withRegion(Regions.US_EAST_1).build();

        //variable declaration for bucket name
        String bucketName = "csci5410-task2-bucket2";

        if (s3.doesBucketExistV2(bucketName))          //checking if bucket already exist
        {
                System.out.format("Bucket %s already exists.\n", bucketName);
        }
        else
        {
                try
                {
                    s3.createBucket(bucketName);      //creating new bucket using createBucket function
                    System.out.println("Successfully bucket creation!");
                }
                catch (AmazonS3Exception e)          //exception handling
                {
                    System.out.println("Error while creating bucket!");
                    System.err.println(e.getErrorMessage());
                }
        }
}}
```


aws
Services
Search for services, features, marketplace products, and docs [Alt+S]
vocstartsoft/user1357694=jn410076@dal.ca @ 4631-1347-4338
Global
Support
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
AWS Marketplace for S3
Provide feedback
Amazon S3
Account snapshot
View Storage Lens dashboard
Copy ARN
Empty
Delete
Create bucket
Find buckets by name

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose **Provide feedback**.

Storage lens provides visibility into storage usage and activity trends. Learn more

Buckets (2)

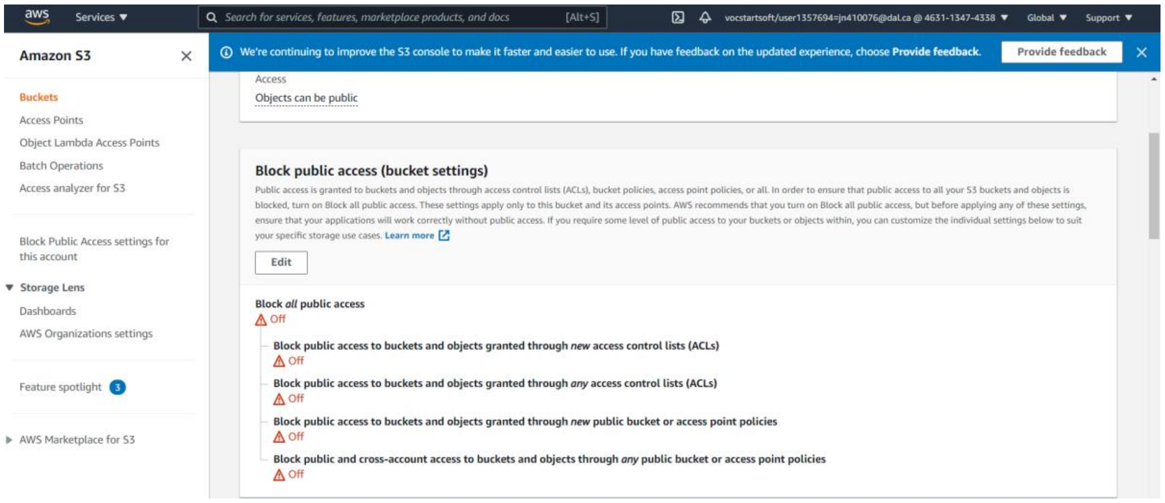
Buckets are containers for data stored in S3. Learn more

Name	AWS Region	Access	Creation date
csci5410-task2-bucket1	US East (N. Virginia) us-east-1	Bucket and objects not public	May 25, 2021, 19:29:46 (UTC-03:00)
csci5410-task2-bucket2	US East (N. Virginia) us-east-1	Objects can be public	May 25, 2021, 20:03:10 (UTC-03:00)

Figure 4: Output for successful bucket creation using java code

b. Disable public access

```
package CSCI5409_Assignment1.AWSproject;

import java.io.IOException;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.PublicAccessBlockConfiguration;
import com.amazonaws.services.s3.model.SetPublicAccessBlockRequest;

public class BlockFileAccess {

        public static void main(String[] args) throws IOException
        {

            //variable declaration for bucket name
            String bucketName = "csci5410-task2-bucket2";
            //connection with AWS s3 using region East 1 and the configurations provided in local folder
            final AmazonS3 s3 = AmazonS3ClientBuilder.standard().withRegion(Regions.US_EAST_1).build();

           //blocking public access
           s3.setPublicAccessBlock(new SetPublicAccessBlockRequest().withBucketName(bucketName)
                           .withPublicAccessBlockConfiguration(new PublicAccessBlockConfiguration()
                           .withBlockPublicAcls(true)
                           .withIgnorePublicAcls(true)
                           .withBlockPublicPolicy(true)
                           .withRestrictPublicBuckets(true)));

           System.out.print("Successfully blocked permission");
         }
}
```


aws
Services ▼
Search for services, features, marketplace products, and docs [Alt+S]
vocstartsoft/user1357694=jn410076@dal.ca @ 4631-1347-4338 ▼
Global ▼
Support ▼
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
AWS Marketplace for S3
Provide feedback
Edit
Learn more

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose **Provide feedback**.

Access
Objects can be public

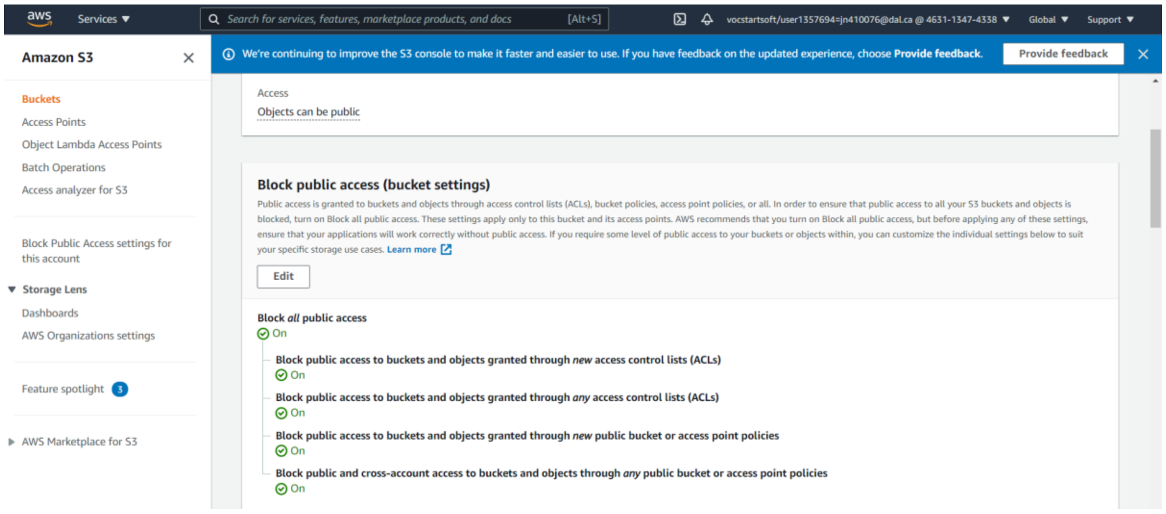
Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

Block *all* public access
⚠ Off

- Block public access to buckets and objects granted through *new* access control lists (ACLs)
  ⚠ Off
- Block public access to buckets and objects granted through *any* access control lists (ACLs)
  ⚠ Off
- Block public access to buckets and objects granted through *new* public bucket or access point policies
  ⚠ Off
- Block public and cross-account access to buckets and objects through *any* public bucket or access point policies
  ⚠ Off

Figure 4: Before running the script for blocking public the permission


aws
Services ▼
Search for services, features, marketplace products, and docs [Alt+S]
vocstartsoft/user1357694=jn410076@dal.ca @ 4631-1347-4338 ▼
Global ▼
Support ▼
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
AWS Marketplace for S3
Provide feedback
Edit
Learn more

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose **Provide feedback**.

Access
Objects can be public

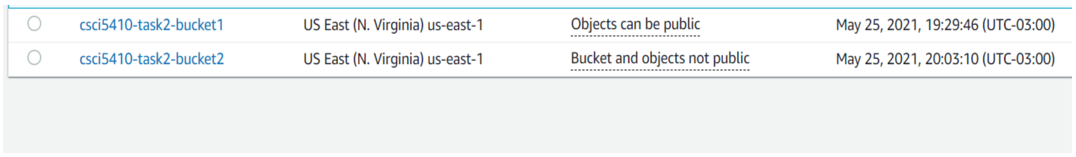
Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

Block *all* public access
✓ On

- Block public access to buckets and objects granted through *new* access control lists (ACLs)
  ✓ On
- Block public access to buckets and objects granted through *any* access control lists (ACLs)
  ✓ On
- Block public access to buckets and objects granted through *new* public bucket or access point policies
  ✓ On
- Block public and cross-account access to buckets and objects through *any* public bucket or access point policies
  ✓ On

Figure 5: After running the script for blocking public the permission

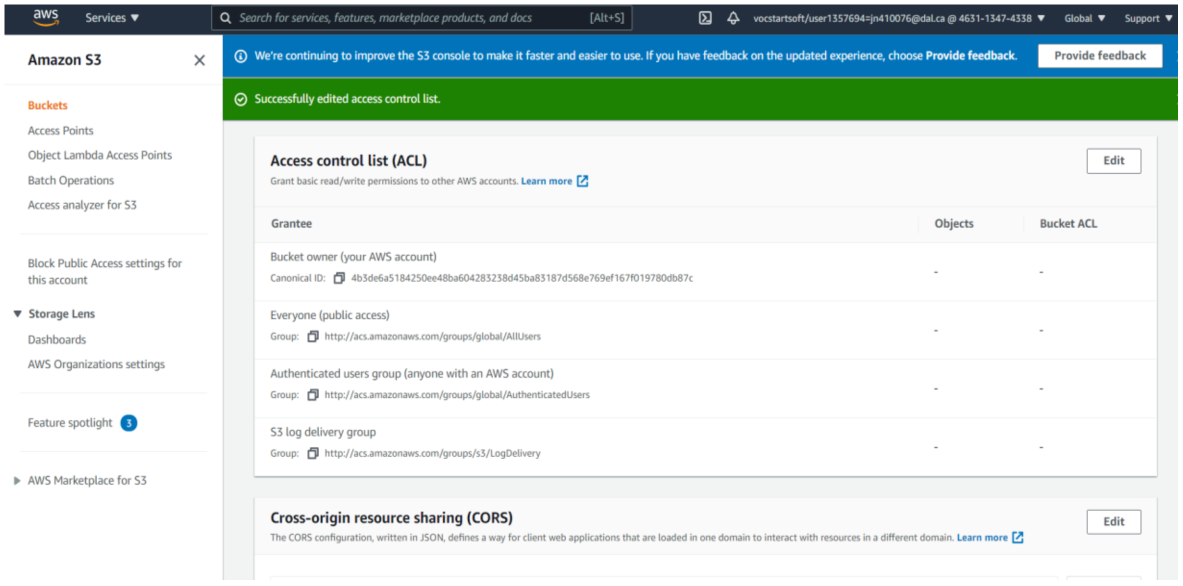


csci5410-task2-bucket1	US East (N. Virginia) us-east-1	Objects can be public	May 25, 2021, 19:29:46 (UTC-03:00)
csci5410-task2-bucket2	US East (N. Virginia) us-east-1	Bucket and objects not public	May 25, 2021, 20:03:10 (UTC-03:00)

Figure 6: Access blocked for bucket2

c. ACL write option to full control for bucket owner

```
package CSCI5409_Assignment1.AWSproject;
import java.io.IOException;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.AccessControlList;
import com.amazonaws.services.s3.model.CanonicalGrantee;
import com.amazonaws.services.s3.model.Grant;
import com.amazonaws.services.s3.model.Permission;

public class ACLFullFileControl
{
        public static void main(String[] args) throws IOException
        {
           //variable declaration for bucket name
           String bucketName = "csci5410-task2-bucket2";

           //connection with AWS s3 using region East 1 and the configurations provided in local folder
           final AmazonS3 s3 = AmazonS3ClientBuilder.standard().withRegion(Regions.US_EAST_1).build();

             final AccessControlList aclcontrol = s3.getBucketAcl(bucketName);

             //grant all permissions
             aclcontrol.grantAllPermissions(new Grant(new CanonicalGrantee(aclcontrol.getOwner().getId()),
                                                                                    Permission.FullControl));
             Grant grant1 = new Grant(new CanonicalGrantee(s3.getS3AccountOwner().getId()),
             Permission.FullControl);

            AccessControlList Bucket = s3.getBucketAcl(bucketName);
            Bucket.grantAllPermissions(grant1);
            s3.setBucketAcl(bucketName, Bucket);
        }
 }
```


aws
Services ▼
Search for services, features, marketplace products, and docs [Alt+S]
vocstartsoft/user1357694=jn410076@dal.ca @ 4631-1347-4338 ▼
Global ▼
Support ▼
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
AWS Marketplace for S3
Provide feedback
Edit
Edit

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose **Provide feedback**.

Successfully edited access control list.

Access control list (ACL)

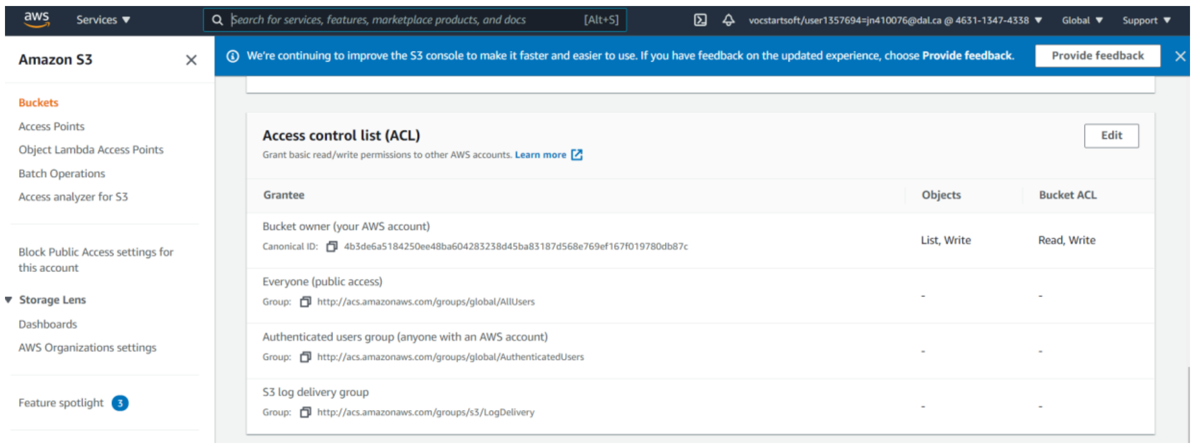
Grant basic read/write permissions to other AWS accounts. Learn more

Grantee	Objects	Bucket ACL
Bucket owner (your AWS account) Canonical ID: 4b3de6a5184250ee48ba604283238d45ba83187d568e769ef167f019780db87c	-	-
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	-	-
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	-	-
S3 log delivery group Group: http://acs.amazonaws.com/groups/s3/LogDelivery	-	-

Cross-origin resource sharing (CORS)

The CORS configuration, written in JSON, defines a way for client web applications that are loaded in one domain to interact with resources in a different domain. Learn more

Figure 7: Changed and removed all access to bucket owner


aws
Services ▼
Search for services, features, marketplace products, and docs [Alt+S]
vocstartsoft/user1357694=jn410076@dal.ca @ 4631-1347-4338 ▼
Global ▼
Support ▼
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
Provide feedback
Edit

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose **Provide feedback**.

Access control list (ACL)

Grant basic read/write permissions to other AWS accounts. Learn more

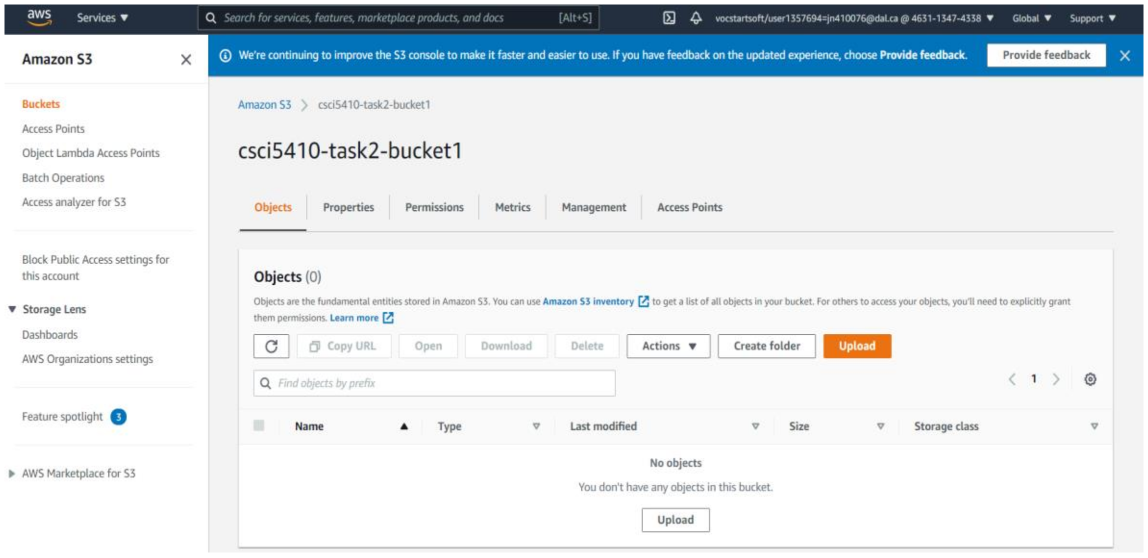
Grantee	Objects	Bucket ACL
Bucket owner (your AWS account) Canonical ID: 4b3de6a5184250ee48ba604283238d45ba83187d568e769ef167f019780db87c	List, Write	Read, Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	-	-
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	-	-
S3 log delivery group Group: http://acs.amazonaws.com/groups/s3/LogDelivery	-	-

Figure 8: After running script for ACL write option to full control for bucket owner

d) Try to move (using your program) the file from 1st bucket to 2nd bucket.

```
package CSCI5409_Assignment1.AWSproject;

import java.io.IOException;
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.CopyObjectRequest;
import com.amazonaws.services.s3.model.DeleteObjectRequest;

public class MoveFile
{
        public static void main(String[] args) throws IOException
```

```
                {
                        //variable declaration for bucket name, file path and file name
                  String sourcebucketName = "csci5410-task2-bucket1";
                  String destinationbucketName = "csci5410-task2-bucket2";
                  String sourceKey = "janvi";
                  String destinationKey = "janvi-bucket2";

                  try
                  {
                    //connection with AWS s3 using region East 1 and the configurations provided in local folder
                    AmazonS3 s3Client =AmazonS3ClientBuilder.standard().withRegion(Regions.US_EAST_1).build();

                    // Copy one object from one bucket to another bucket and delete it from source bucket
                    CopyObjectRequest copyObjRequest = new CopyObjectRequest(sourcebucketName, sourceKey,
                                              destinationbucketName, destinationKey);
                    s3Client.copyObject(copyObjRequest);
                    s3Client.deleteObject(new DeleteObjectRequest(sourcebucketName, sourceKey));
                  }
                  catch(AmazonServiceException e)
                  {
                    // The call was transmitted successfully, but Amazon S3 couldn't process
                    // it, so it returned an error response.
                    e.printStackTrace();
                  }
                  catch(SdkClientException e) {
                    // Amazon S3 couldn't be contacted for a response, or the client
                    // couldn't parse the response from Amazon S3.
                    e.printStackTrace();
                  }
                 }
            }
```



We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose **Provide feedback**.

csci5410-task2-bucket1

Objects (0)

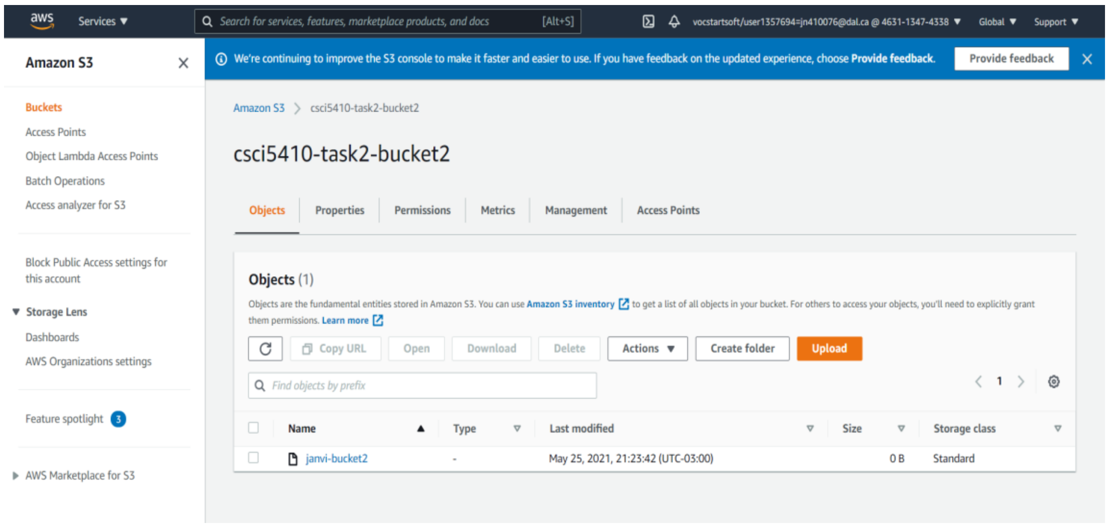
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class

No objects

You don't have any objects in this bucket.

Figure 9: janvi.txt is removed from bucket 1



We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose **Provide feedback**.

csci5410-task2-bucket2

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use **Amazon S3 inventory** to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. **Learn more**

Name	Type	Last modified	Size	Storage class
janvi-bucket2	-	May 25, 2021, 21:23:42 (UTC-03:00)	0 B	Standard

Figure 10: janvi.txt from bucket1 is shifted to bucket2

Flowchart:

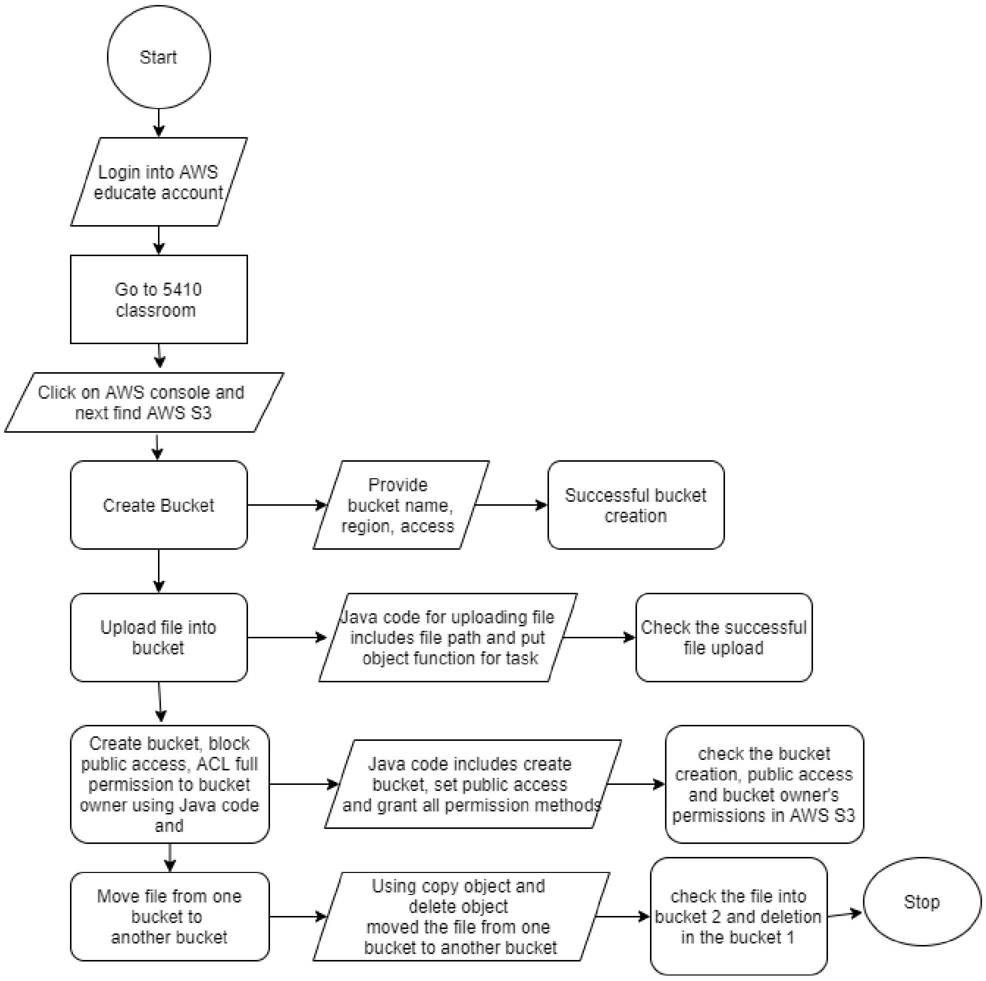

Start
Login into AWS educate account
Go to 5410 classroom
Click on AWS console and next find AWS S3
Create Bucket
Provide bucket name, region, access
Successful bucket creation
Upload file into bucket
Java code for uploading file includes file path and put object function for task
Check the successful file upload
Create bucket, block public access, ACL full permission to bucket owner using Java code and
Java code includes create bucket, set public access and grant all permission methods
check the bucket creation, public access and bucket owner's permissions in AWS S3
Move file from one bucket to another bucket
Using copy object and delete object moved the file from one bucket to another bucket
check the file into bucket 2 and deletion in the bucket 1
Stop

Figure 11: AWS S3 storage experiment

A paragraph on your overall observation:

AWS S3 storage experiment provided idea on how buckets are being created with some properties such as region, access permission and storage class. Overall the exercise involved different operations – create, upload and move which was being done by some java s3 functions:

- Createbucket using same funciton
- Move file to other bucket copybucket, deletebucket
- Uploadfile to bucket : putbucket

Access control has also been performed here and which can be changed by AWS S3 console as well as JAVA code. I have also observed that there are some user rights read, write and list which can be provided or changed through out the use of bucket.

- Block public access for bucket: setPublicAccessBlock
- Grant full access to user for bucket: grantAllPermission

References:
[1] https://docs.aws.amazon.com/AmazonS3/latest/userguide/configuring-block-public-access-bucket.html

[2] https://docs.aws.amazon.com/AmazonS3/latest/userguide/managing-acls.html

[3] https://github.com/awsdocs/aws-doc-sdk examples/blob/master/javav2/example_code/s3/src/main/java/com/example/s3/SetAcl.java

[4] https://docs.aws.amazon.com/AmazonS3/latest/userguide/creating-bucket.html

[5] https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/examples-s3-buckets.html