ASSIGNMENT 1: PART B

TASK A: Flowchart of the work.

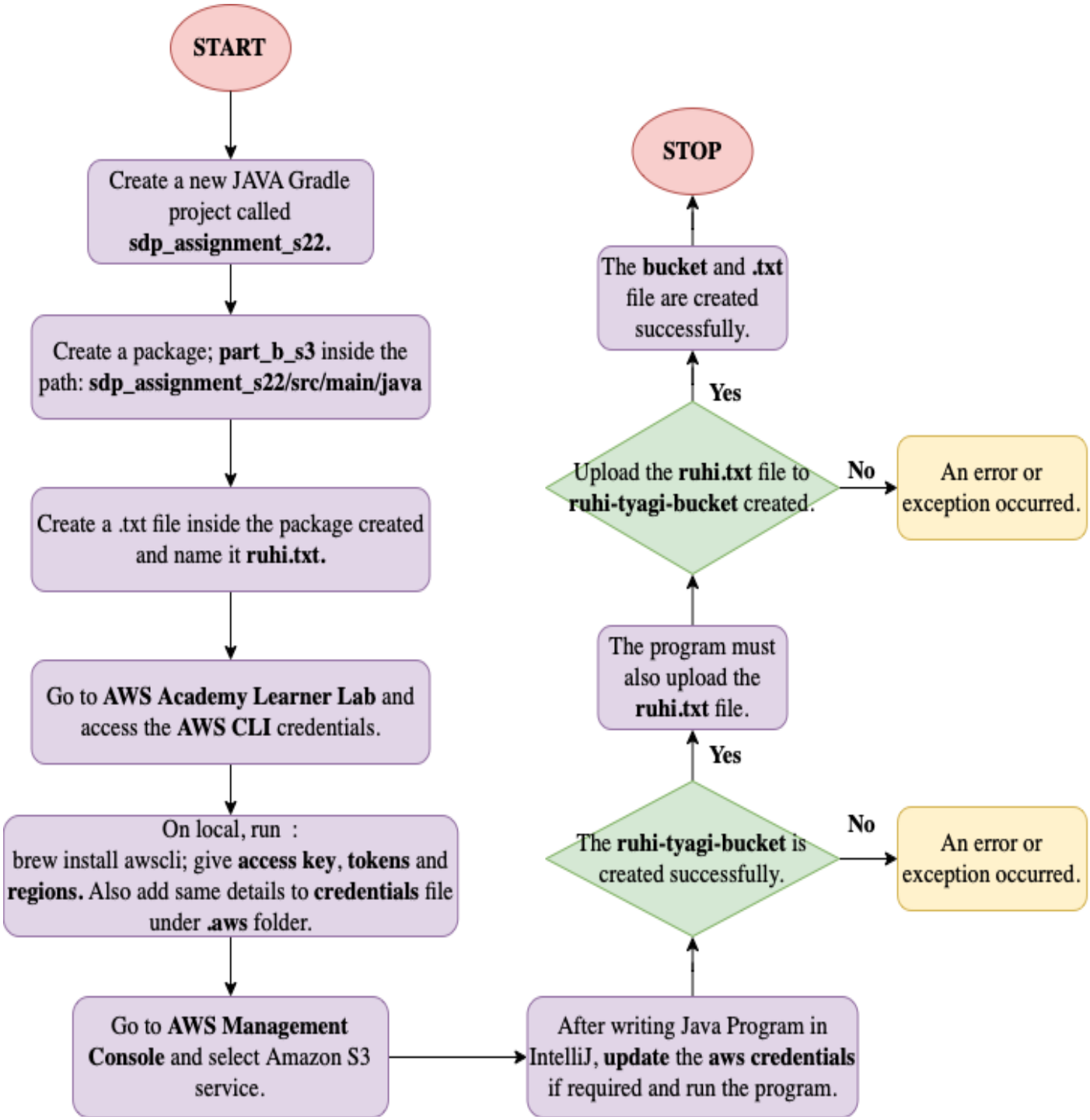

Figure 1. *Flowchart of part B.*

TASK B: A paragraph on overall observation of AWS SDK for Java.

The AWS SDK for Java provides a Java API for AWS services. The SDK allows users to quickly construct Java apps that interact with Amazon S3, Amazon EC2, DynamoDB, and other services [1]. Gradle's improved POM support capability can be used to import BOM files by defining a reliance on a BOM, which I did because I built the project with Gradle v5.0+. I was using Gradle 7.4.1, and I didn't have to use the enableFeaturePreview command; instead, I just implemented the dependency directly. The ease with which users may connect to and use AWS's services was a major benefit of the AWS SDK for Java. A developer can use a range of built-in classes and methods to manipulate the services. I used two in-built methods where the first of which was to create a bucket using the createBucket() method, which made things a lot easier because all I had to do was provide the bucketName and the bucket was formed in AmazonS3. Similarly, with the second method, I can use the putObject() function to upload a file to that bucket or any other existing bucket, which requires the bucket name, file name, and the file to be uploaded. As a result, using the AWS SDK for Java made it extremely easy to implement any concept as well as add more capabilities without having to write the boilerplate code.

TASK C: Screenshots of all the steps performed in creating bucket and uploading txt file in Amazon S3.

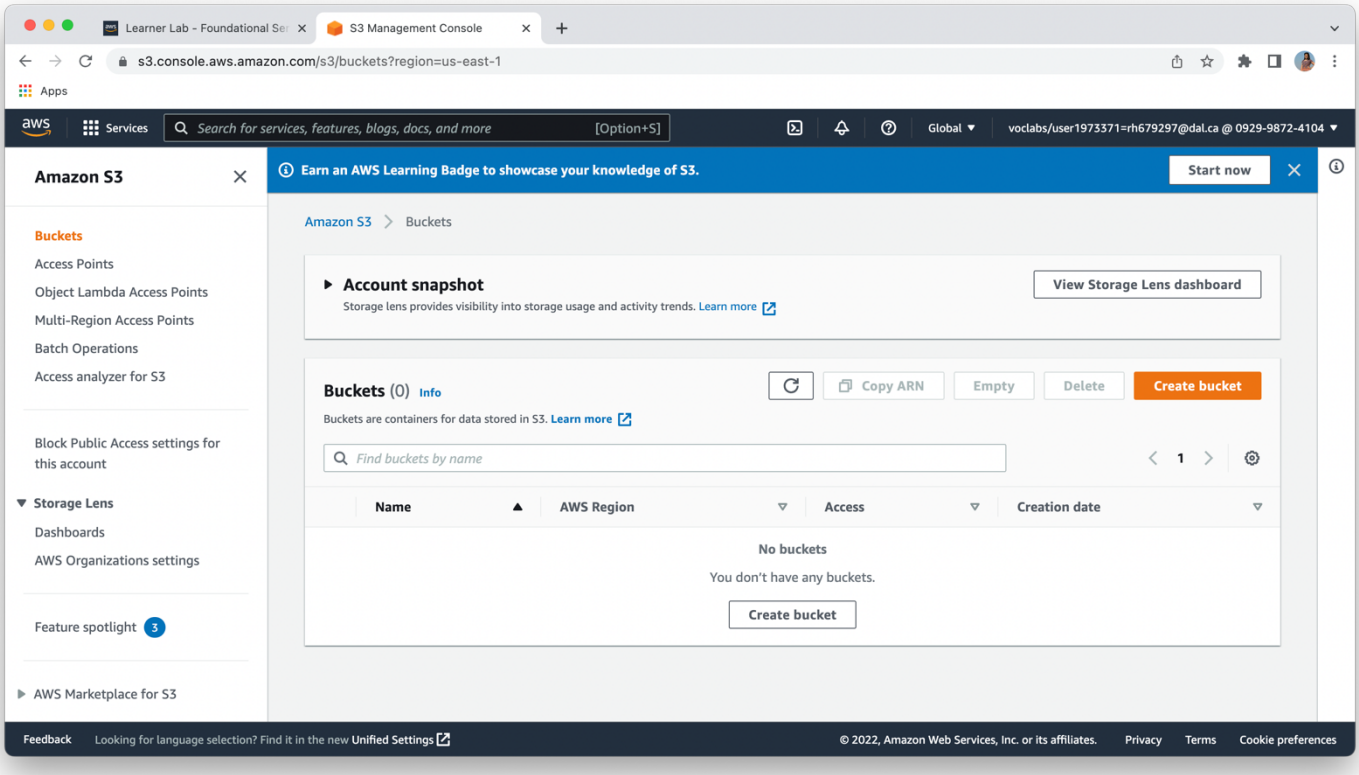

Figure 1. *Screenshot of the Amazon S3 service BEFORE creating the bucket.*

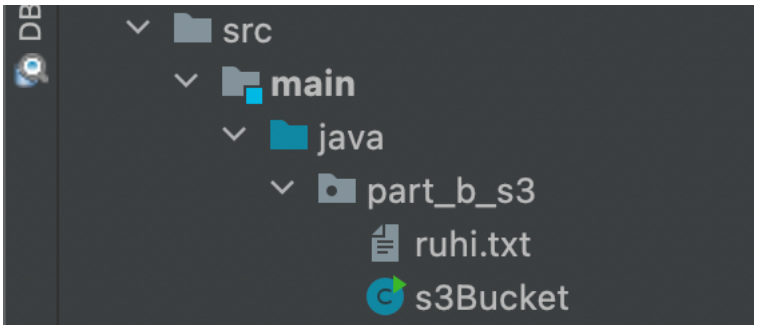

Figure 2. *Screenshot of part B package from IntelliJ.*

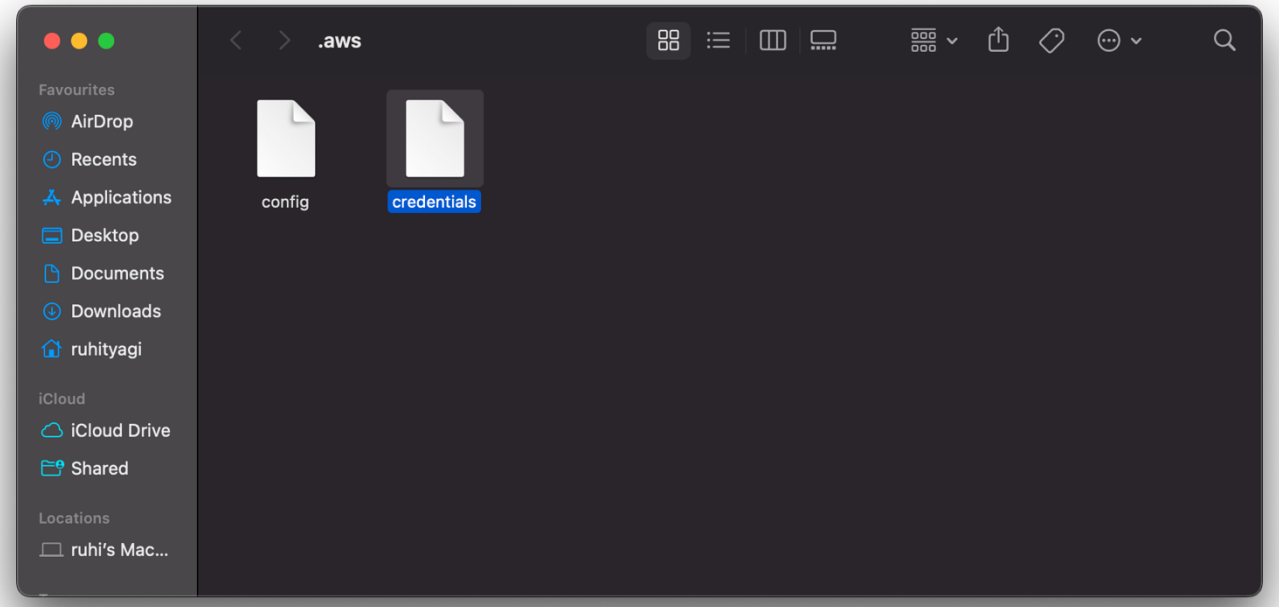

Figure 3. *Screenshot of .aws folder.*

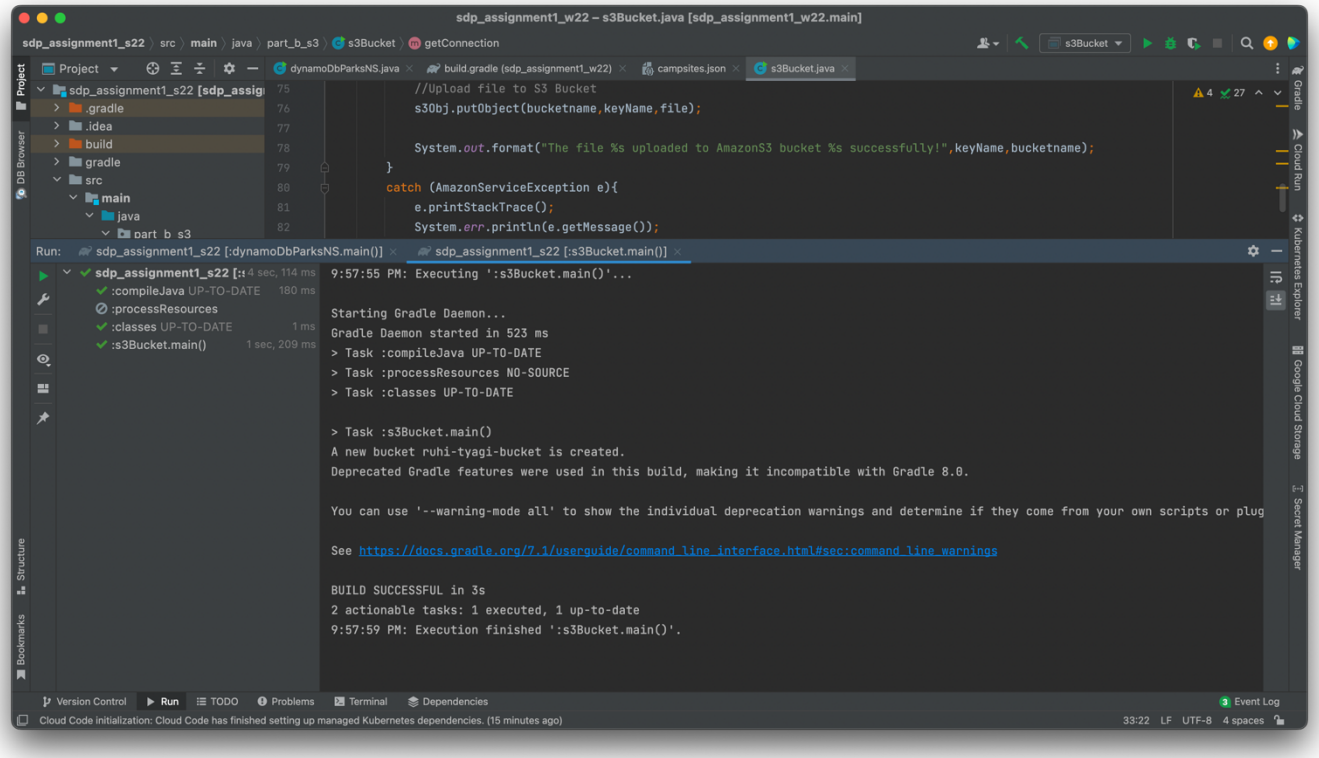

Figure 4. *Screenshot of successful creation of ruhi-tyagi-bucket by JAVA program [2].*

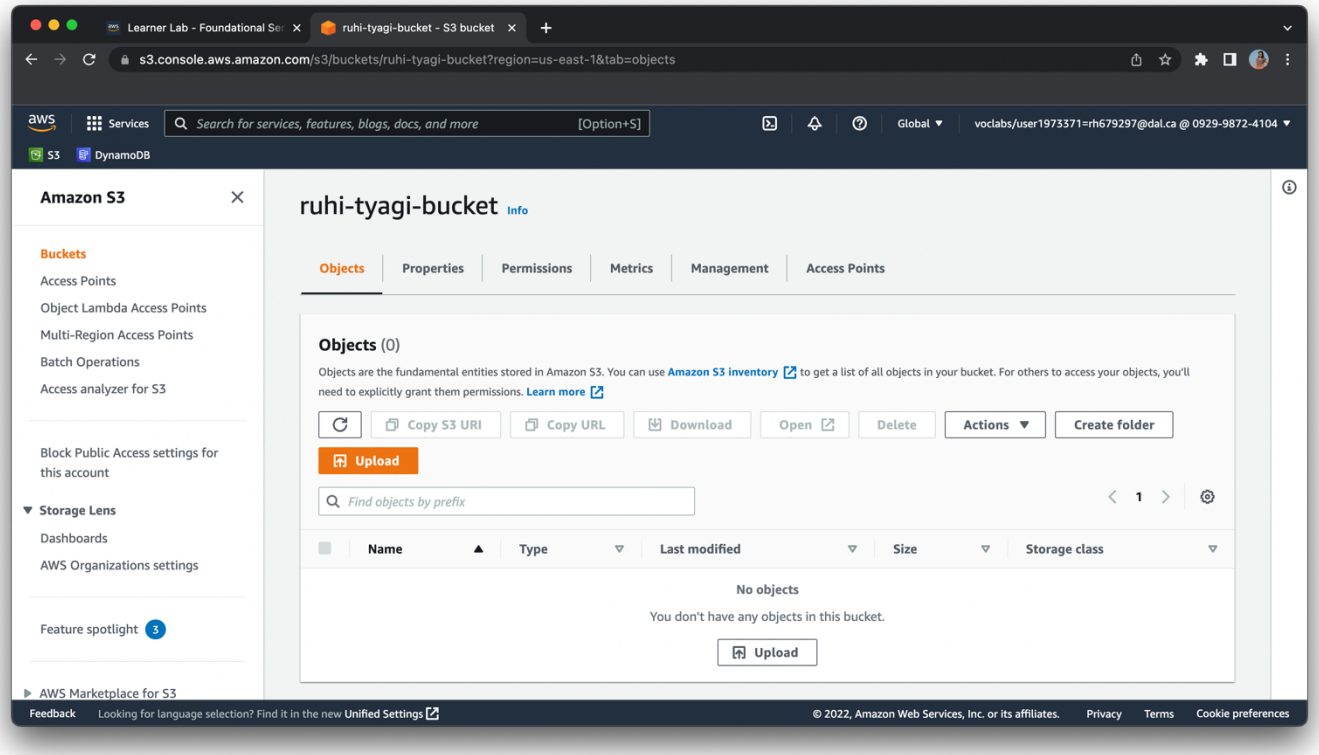

Figure 5. *Screenshot of empty bucket.*

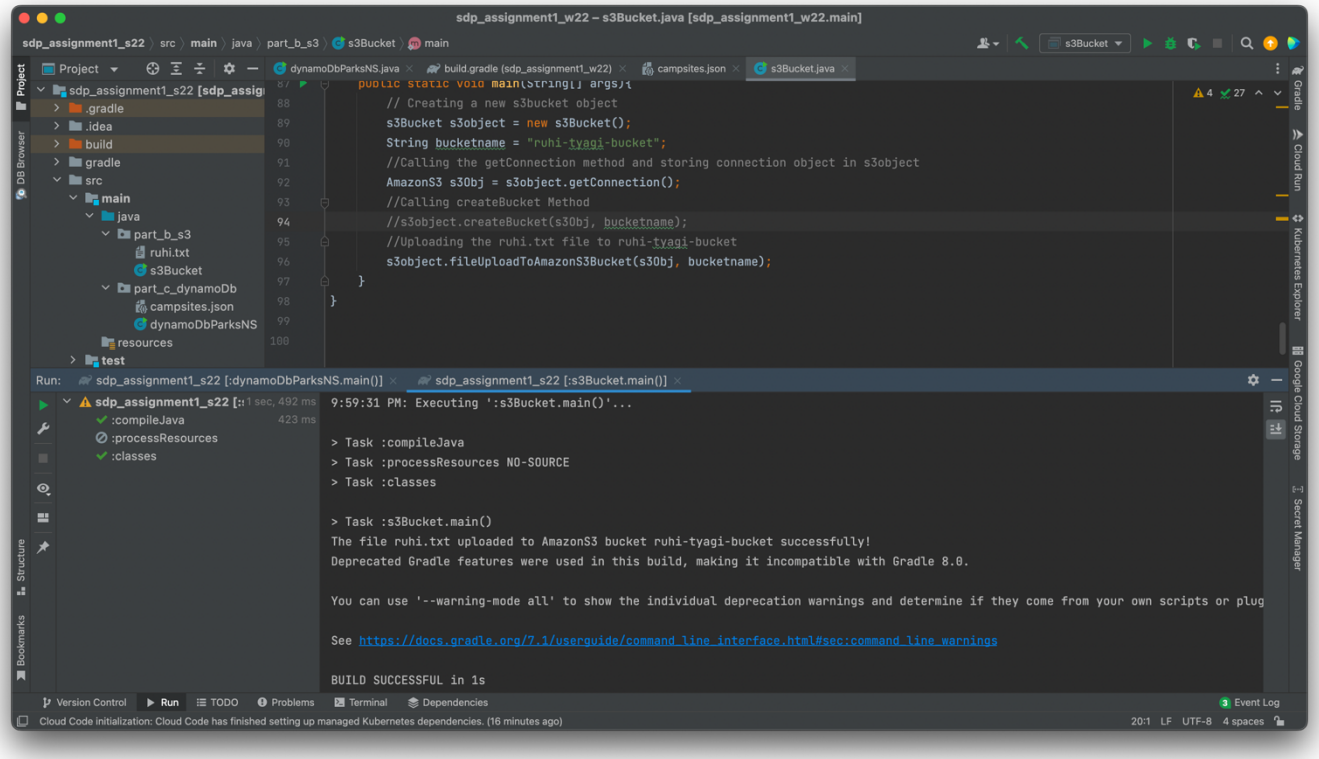

Figure 6. *Screenshot of successful upload of ruhi.txt file in IntelliJ output [3].*

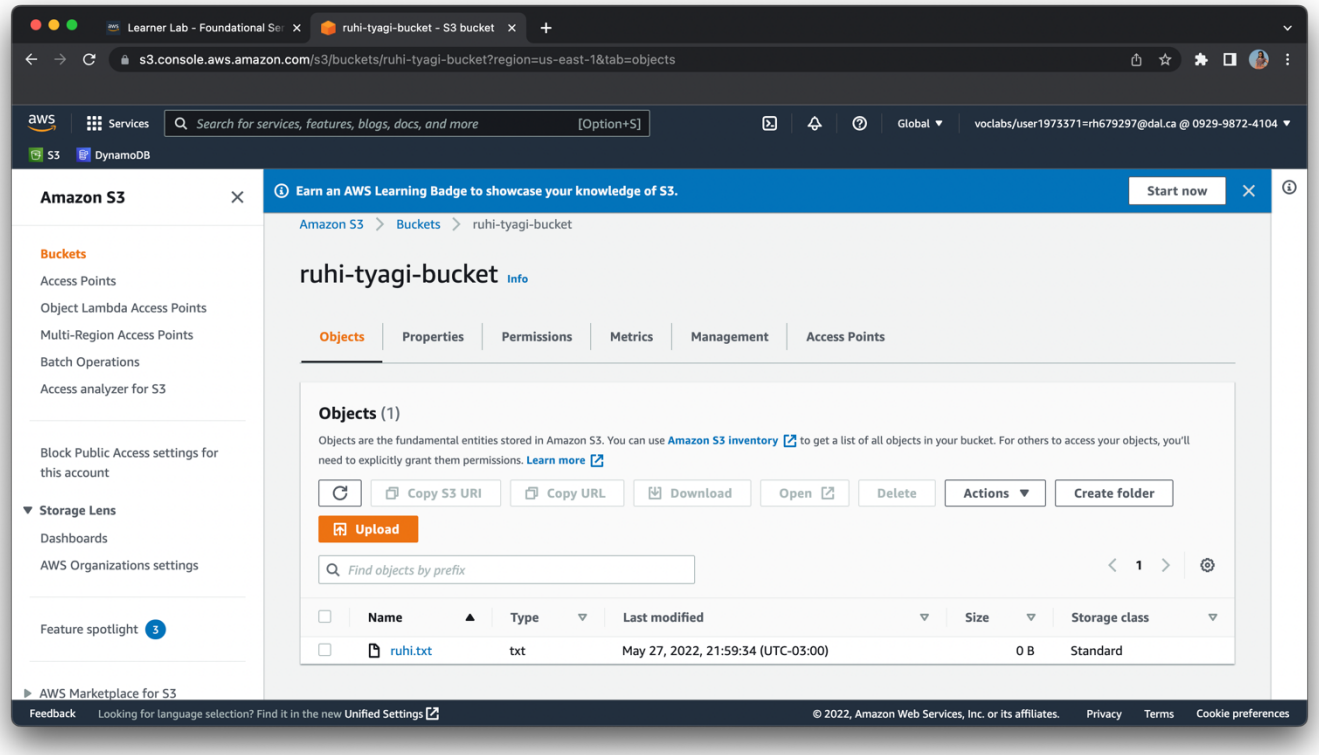

Figure 7. *Screenshot of successful upload of ruhi.txt file on Amazon S3.*

TASK D: Code snapshot.

***s3Bucket.java*:**

```
package part_b_s3;
/**
 * Author: Ruhi Rajnish Tyagi
 * Banner Id: B00872269
 */

import com.amazonaws.AmazonServiceException;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.BasicSessionCredentials;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.AmazonS3Exception;
import com.amazonaws.services.s3.model.Bucket;

import java.io.File;
import java.nio.file.Paths;

public class s3Bucket {

    //method to connect to the AWS account
    public AmazonS3 getConnection(){

        //provide credentials to create connection
        BasicSessionCredentials sessionCredentials = new BasicSessionCredentials(
                "<access-key>",
                "<secret-key",
               "<session-token>"
                );

        //Connect to AmazonS3 service and store in an object
        AmazonS3 s3Obj = AmazonS3ClientBuilder.standard().withCredentials
                (new AWSStaticCredentialsProvider(sessionCredentials))
                .withRegion(Regions.US_EAST_1)
                .build();

        return s3Obj;
    }

    // The public method to create a new S3 bucket
    public void createBucket (AmazonS3 s3Obj, String bucketname) {
        Bucket b = null;
        //Specify bucketname as per rules
        //check if bucket exists otherwise create new.
        if (s3Obj.doesBucketExistV2(bucketname)) {
            System.out.format("Bucket %s already exists.", bucketname);
        }
        else {
            try{
                s3Obj.createBucket (bucketname);
                System.out.format ("A new bucket %s is created.",bucketname);
            }
            catch (AmazonS3Exception e) {
                e.printStackTrace();
                System.err.println(e.getMessage());
            }
        }
    }
```

```
    //method to upload txt file to S3bucket
    public void fileUploadToAmazonS3Bucket(AmazonS3 s3Obj,
                                           String bucketname){
        try {
            //File path to txt file
            String filepath ="./src/main/java/part_b_s3/ruhi.txt";
            //Create new file with passed filename
            File file = new File(filepath);
            //extract filename and store it in a variable
            String keyName = Paths.get(filepath).getFileName().toString();
            //Upload file to S3 Bucket
            s3Obj.putObject(bucketname,keyName,file);

            System.out.format("The file %s uploaded to AmazonS3 bucket %s
successfully!",keyName,bucketname);
        }
        catch (AmazonServiceException e){
            e.printStackTrace();
            System.err.println(e.getMessage());
        }

    }

    public static void main(String[] args){
        // Creating a new s3bucket object
        s3Bucket s3object = new s3Bucket();
        String bucketname = "ruhi-tyagi-bucket";
        //Calling the getConnection method and storing connection object in s3object
        AmazonS3 s3Obj = s3object.getConnection();
        //Calling createBucket Method
        s3object.createBucket(s3Obj, bucketname);
        //Uploading the ruhi.txt file to ruhi-tyagi-bucket
        s3object.fileUploadToAmazonS3Bucket(s3Obj, bucketname);
    }
}
```

Link to gitlab: https://git.cs.dal.ca/rtyagi/csci5410_b00872269_ruhirajnish_tyagi/-/tree/main/src/main/java/part_b_s3

REFERENCES:

[1] "Developer guide - AWS SDK for Java 2.x", *AWS*, 2022. [Online]. Available: https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/home.html. [Accessed: 25- May- 2022]

[2] "Creating, Listing, and Deleting Amazon S3 Buckets", *AWS*, 2022. [Online]. Available: https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/examples-s3-buckets.html. [Accessed: 25- May- 2022]

[3] "Uploading objects", *AWS*, 2022. [Online]. Available: https://docs.aws.amazon.com/AmazonS3/latest/userguide/upload-objects.html. [Accessed: 25- May- 2022]