

1. Create an S3 bucket and upload some objects to S3.

- go to aws console--search s3--create a bucket
- bucket--unique name--disable acl---disable block public access--create
- upload objects--

The screenshot shows the AWS Management Console interface for S3 buckets. The top section is titled 'General purpose buckets (1)' and includes a search bar and buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'. Below this, a table lists the bucket 'lounge-template' in the 'US East (N. Virginia) us-east-1' region, created on 'September 29, 2025, 15:23:59 (UTC+05:30)'. The bottom section shows the 'Objects' page for the 'lounge-template' bucket, displaying a list of 13 objects. The objects include various image files (png, ico) and folders (css/, images/).

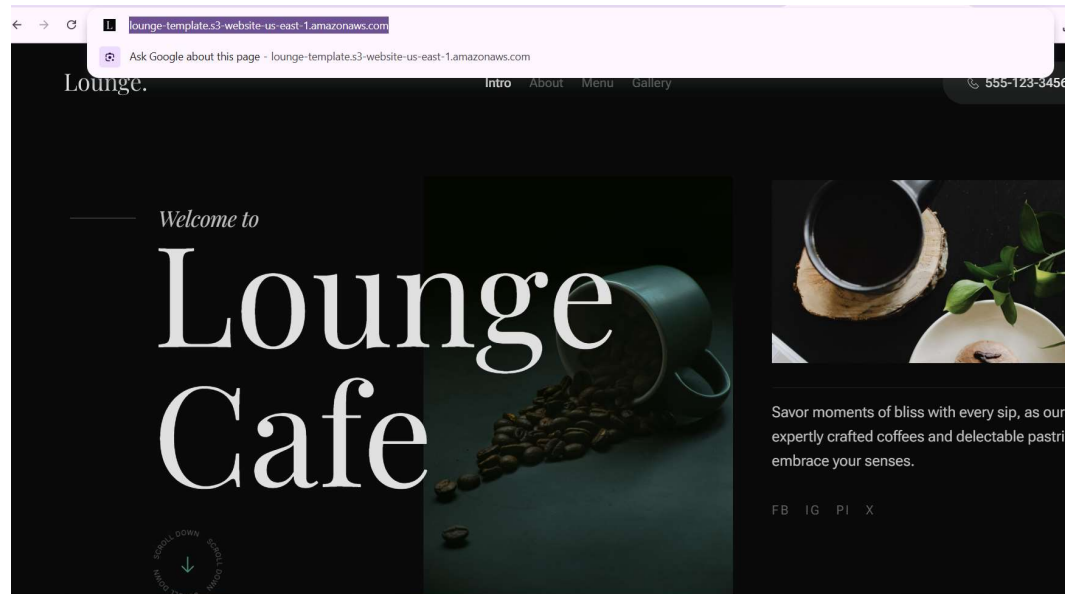
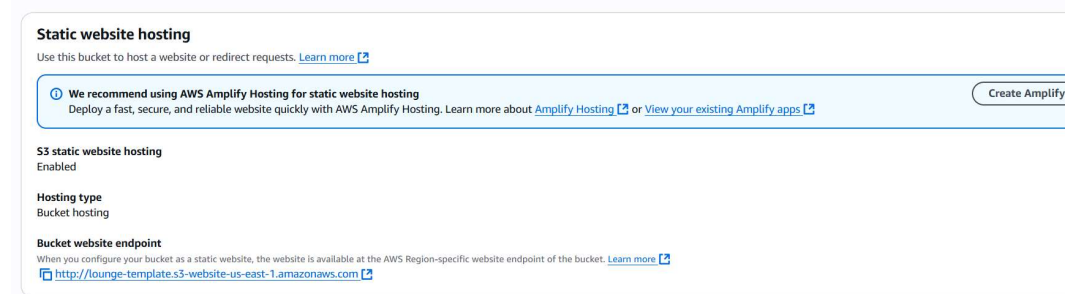
| Name | Type | Last modified | Size | Storage class |
|----------------------------|--------|--|----------|---------------|
| android-chrome-192x192.png | png | September 29, 2025, 15:36:11 (UTC+05:30) | 5.1 KB | Standard |
| android-chrome-512x512.png | png | September 29, 2025, 15:36:11 (UTC+05:30) | 16.6 KB | Standard |
| apple-touch-icon.png | png | September 29, 2025, 15:36:11 (UTC+05:30) | 4.6 KB | Standard |
| css/ | Folder | - | - | - |
| favicon-16x16.png | png | September 29, 2025, 15:36:12 (UTC+05:30) | 511.0 B | Standard |
| favicon-32x32.png | png | September 29, 2025, 15:36:12 (UTC+05:30) | 1023.0 B | Standard |
| favicon.ico | ico | September 29, 2025, 15:36:16 (UTC+05:30) | 15.0 KB | Standard |
| images/ | Folder | - | - | - |

2. Deploy a static website in the S3 bucket.

- make bucket as public by making acl disable and remove block

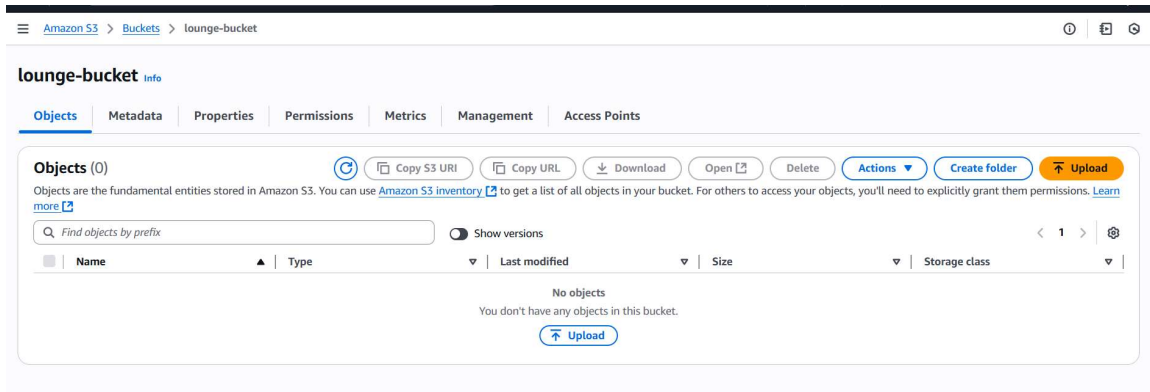
public access.

- go to your bucket---properties---enable static website---index.html-
- make objects as public---go to objects--actions--make objects public

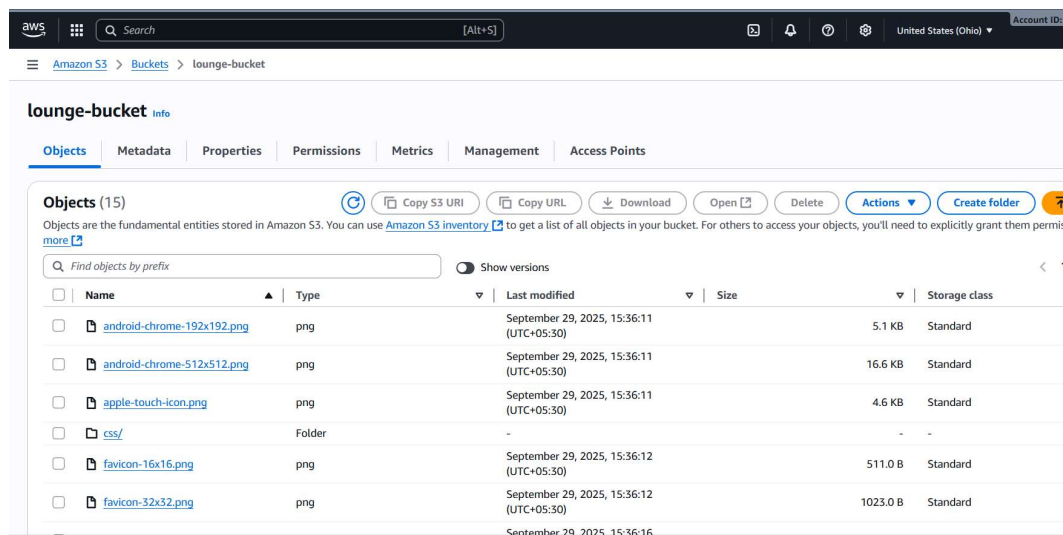


3. Enable cross-region replication on S3 buckets.

- create a bucket in any region



- then go to bucket in n.virginia region
lounge-bucket---management---create replication rules---source bucket---destination bucket (other bucket in any region or any account)---define replication rules to all objects in the bucket or limit the objects---create
- create a job ---create an iam role---schedule time ---define conditions to replicate(replicate when job run assigned or replicate all time)
- now you can check the other bucket the objects will be replicated



4. Configure a bucket policy so only the Admin user can see the

objects of the S3 bucket.

- create a bucket in with object (index.html)
- create an I'm user and attach policies s3 full access then login with that user
- then go to your bucket permissions and paste these code in bucket policy with username,account id and s3 bucket name
- now try to access that bucket from other user.
- you wont be able to access the bucket .
- user-1 view

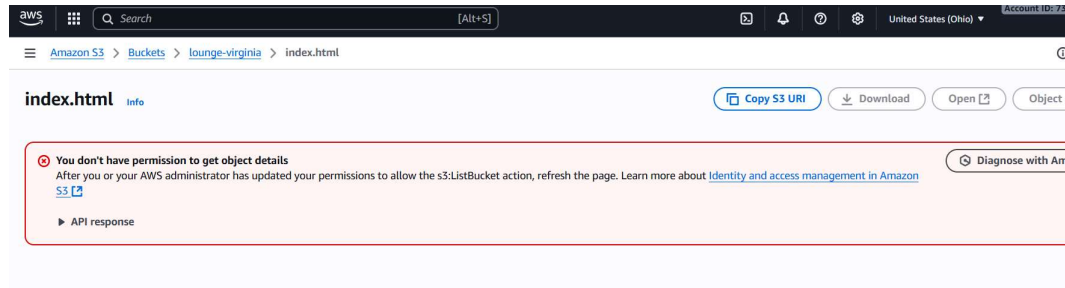
Bucket policy

Edit

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

```
{
  "Version": "2012-10-17",
  "Id": "LoungeVirginiaPolicy",
  "Statement": [
    {
      "Sid": "AllowDevOpsReadAccess",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::734846753465:user/devops"
      },
      "Action": [
        "s3:GetObject",
        "s3:GetBucketLocation",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::lounge-virginia/*",
        "arn:aws:s3:::lounge-virginia"
      ]
    }
  ],
}
```

- user-2



-
-
- Go to the S3 console
- Select your bucket
- Navigate to the "Permissions" tab
- Scroll to "Bucket policy"
- Click "Edit" and paste this policy
- Click "Save changes"
-
- Important Notes:
-
- This policy uses NotPrincipal with Deny which is very restrictive and will block all other users including root
- If you need the root user to also have access, you should adjust the policy
- Make sure the Admin user exists before applying this policy
- paste the code in bucket policy

```

{
  "Version": "2012-10-17",
  "Id": "LoungeVirginiaPolicy",
  "Statement": [
    {
      "Sid": "AllowDevOpsReadAccess",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::734846753465:user/devops"
      },
      "Action": [
        "s3:GetObject",
        "s3:GetBucketLocation",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::lounge-virginia/*",
        "arn:aws:s3:::lounge-virginia"
      ]
    },
    {
      "Sid": "DenyAllOthers",
      "Effect": "Deny",
      "NotPrincipal": {
        "AWS": "arn:aws:iam::734846753465:user/devops"
      }
    }
  ]
}

```

```

    },
    "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:ListBucket"
    ],
    "Resource": [
        "arn:aws:s3:::lounge-virginia/*",
        "arn:aws:s3:::lounge-virginia"
    ]
}
]
}

```

5. Set up lifecycle policies to automatically transition or delete objects based on specific criteria.

us-east-2.console.aws.amazon.com/s3/management/new-bucket19196/lifecycle/edit?region=us-east-2&bucketType=general&id=new-rule

Amazon S3 > Buckets > new-bucket19196 > Lifecycle configuration > new-rule

Lifecycle rule name

new-rule
Up to 255 characters

Choose a rule scope

☐ Limit the scope of this rule using one or more filters

☒ Apply to all objects in the bucket

⚠ Apply to all objects in the bucket
If you want the rule to apply to specific objects, you must use a filter to identify those objects. Choose "Limit the scope of this rule using one or more filters". [Learn more](#)

☒ I acknowledge that this rule will apply to all objects in the bucket.

Lifecycle rule actions
Choose the actions you want this rule to perform.

☒ Transition current versions of objects between storage classes
This action will move current versions.

☐ Transition noncurrent versions of objects between storage classes
This action will move noncurrent versions.

☐ Expire current versions of objects

☐ Permanently delete noncurrent versions of objects

☒ Delete expired object delete markers or incomplete multipart uploads
These actions are not supported when filtering by object tags or object size.

By default, objects less than 128KB will not transition across any storage class. We don't recommend transitioning objects less than 128 KB because the transition costs can outweigh the storage savings. If your use case requires transitioning objects less than 128 KB, specify a minimum object size filter for each applicable lifecycle rule with a transition action.

Transition current versions of objects between storage classes
Choose transitions to move current versions of objects between storage classes based on your use case scenario and performance access requirements. These transitions start from when the objects are created and are consecutively applied. [Learn more](#)

Choose storage class transitions: Standard-IA
Days after object creation: 30
[Remove](#)
[Add transition](#)

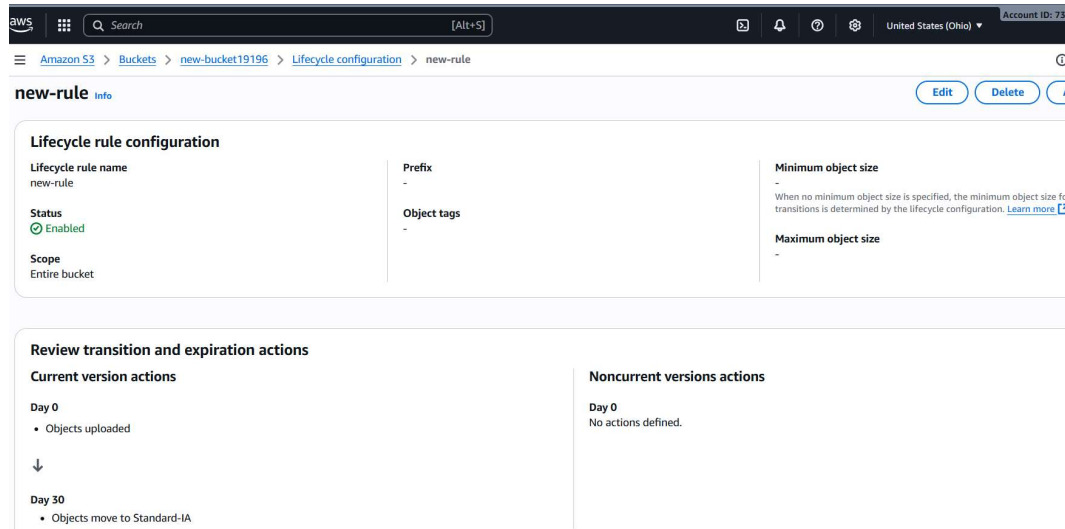
Delete expired object delete markers or incomplete multipart uploads
Expired object delete markers
This action will remove expired object delete markers and may improve performance. An expired object delete marker is removed if all noncurrent versions of an object expire after deleting a versioned object. This action is not available when "Expire versions of objects" is selected. [Learn more](#)
☐ Delete expired object delete markers
Incomplete multipart uploads
This action will stop all incomplete multipart uploads, and the parts associated with the multipart upload will be deleted. [Learn more](#)
☐ Delete incomplete multipart uploads

Delete expired object delete markers or incomplete multipart uploads
Expired object delete markers
This action will remove expired object delete markers and may improve performance. An expired object delete marker is removed if all noncurrent versions of an object expire after deleting a versioned object. This action is not available when "Expire versions of objects" is selected. [Learn more](#)
☒ Delete expired object delete markers
Incomplete multipart uploads
This action will stop all incomplete multipart uploads, and the parts associated with the multipart upload will be deleted. [Learn more](#)
☐ Delete incomplete multipart uploads

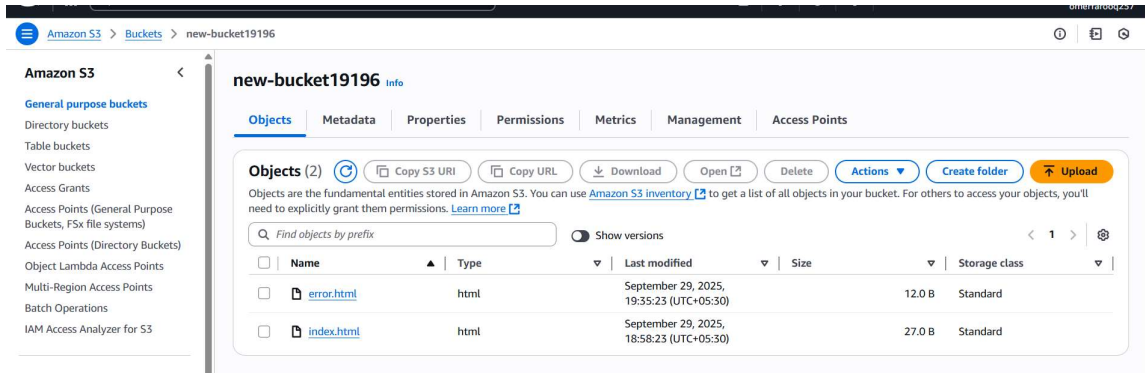
Review transition and expiration actions
Current version actions
Day 0
• Objects uploaded
↓
Day 30
• Objects move to Standard-IA
Noncurrent versions actions
Day 0
No actions defined.

- go to bucket management---create life cycle rules---select life rule actions---acknowledge for charges---select storage class and no. of days---and create
- Go to bucket → Management tab
- Click Lifecycle rules → Create lifecycle rule
- Enter rule name
- Choose rule scope (entire bucket or prefix)
- Add lifecycle rule actions:

-
- Transition current versions
- Transition previous versions
- Expire current versions
- Delete expired object delete markers
-
-
- Set days and storage classes
- Click Create rule



6. Push some objects to S3 using the AWS CLI.



```

MINGW64:/c/Users/DELL/Downloads

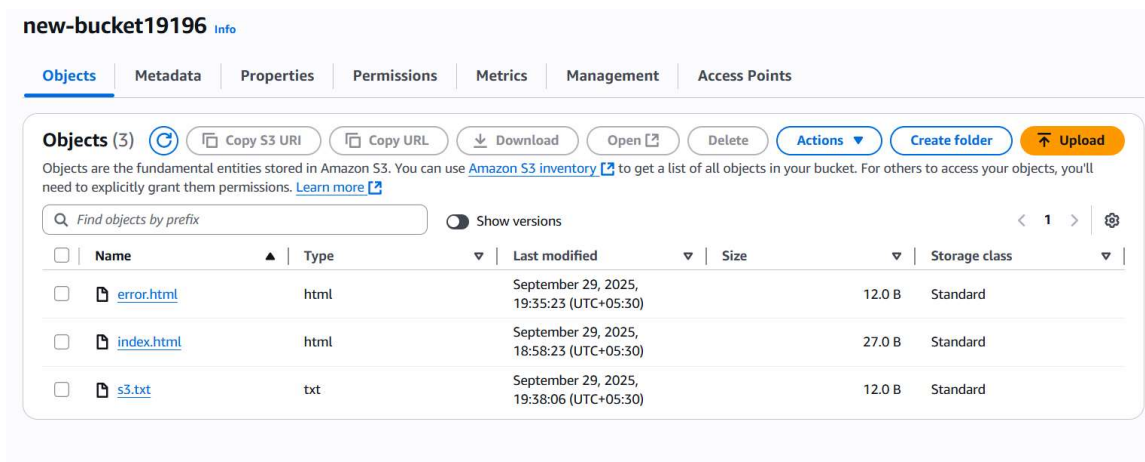
DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ echo "hello world">s3.txt

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ cat s3.txt
hello world

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ aws s3 cp s3.txt s3://new-bucket19196/
upload: .\s3.txt to s3://new-bucket19196/s3.txt

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$

```



- `aws s3 cp filename s3://bucket-name/` (to push files from cli to s3 bucket)

7. Upload a 1 GB file to S3 using the CLI.

```
MINGW64:/c/Users/DELL/Downloads

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ aws s3 ls
2025-09-29 18:58:02 new-bucket19196

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ aws s3 ls new-bucket19196
2025-09-29 19:35:23      12 error.html
2025-09-29 18:58:23      27 index.html
2025-09-29 19:38:06      12 s3.txt

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ aws s3 cp s3-bucket-policy.mp4 s3://new-bucket19196/
Completed 33.0 MiB/415.5 MiB (4.3 MiB/s) with 1 file(s) remaining
```

new-bucket19196 [Info](#)

[Objects](#) | [Metadata](#) | [Properties](#) | [Permissions](#) | [Metrics](#) | [Management](#) | [Access Points](#)

Objects (4) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

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](#)

☒ Show versions < 1 > [Settings](#)

| <input type="checkbox"/> | Name | Type | Last modified | Size | Storage class |
|--------------------------|--------------------------------------|------|--|----------|---------------|
| <input type="checkbox"/> | error.html | html | September 29, 2025, 19:35:23 (UTC+05:30) | 12.0 B | Standard |
| <input type="checkbox"/> | index.html | html | September 29, 2025, 18:58:23 (UTC+05:30) | 27.0 B | Standard |
| <input type="checkbox"/> | s3-bucket-policy.mp4 | mp4 | September 29, 2025, 19:50:49 (UTC+05:30) | 415.5 MB | Standard |
| <input type="checkbox"/> | s3.txt | txt | September 29, 2025, 19:38:06 (UTC+05:30) | 12.0 B | Standard |

```
DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ aws s3 rm s3://new-bucket19196/s3-bucket-policy.mp4
delete: s3://new-bucket19196/s3-bucket-policy.mp4

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ aws s3 ls new-bucket19196
2025-09-29 19:35:23      12 error.html
2025-09-29 18:58:23      27 index.html
2025-09-29 19:38:06      12 s3.txt
```

to copy 1gb file

aws s3 ls (to see list of bucket)

aws s3 ls bucket-name (to see list of objects in a bucket)

aws s3 cp file-name s://bucket-name (to copy large files to bucket using cli)

aws s3 rm s://bucket-name/file-name (to remove any file from bucket using cli)

8. . Write a Bash script to create an S3 bucket

#!/bin/bash

```
DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ vi s3.sh

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ sh s3.sh
make_bucket: saif-bucket-19196
Bucket 's3://saif-bucket-19196' created successfully in us-east-1

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$ cat s3.sh
BUCKET_NAME="saif-bucket-19196"
REGION="us-east-1"

# Create the S3 bucket in a specific region
aws s3 mb s3://$BUCKET_NAME --region $REGION

echo "Bucket 's3://$BUCKET_NAME' created successfully in $REGION"

DELL@DESKTOP-RCNKJC4 MINGW64 ~/Downloads
$
```

BUCKET_NAME="saif-bucket-19196"

```
REGION="us-east-1"
```

```
# Create the S3 bucket in a specific region
```

```
aws s3 mb s3://$BUCKET_NAME --region $REGION
```

```
echo "Bucket 's3://$BUCKET_NAME' created successfully in $REGION"
```