

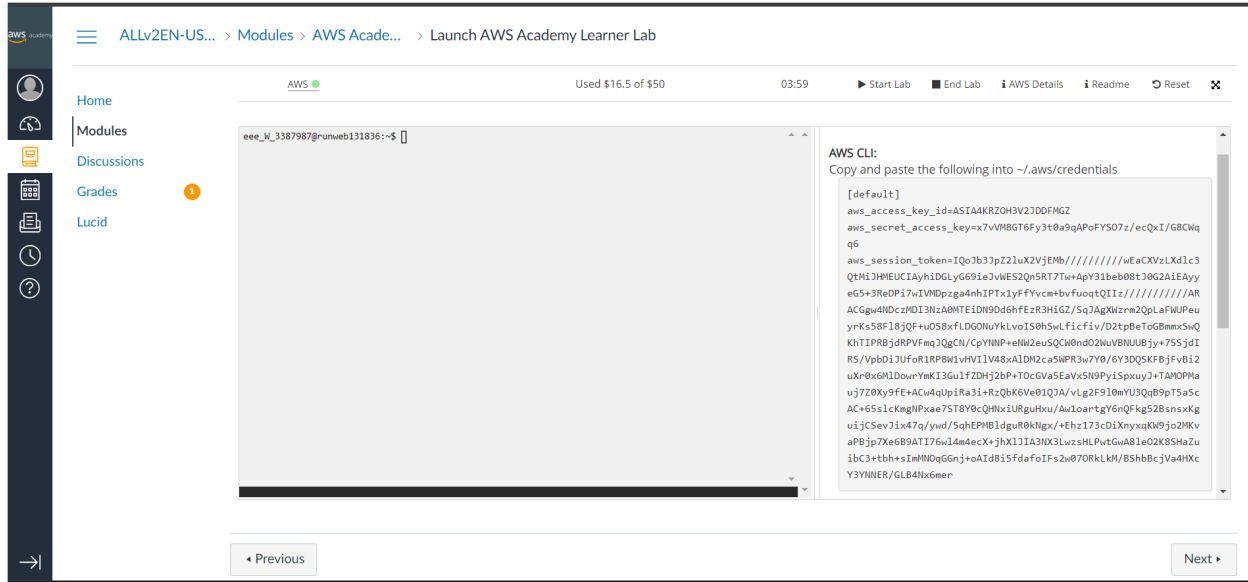
Aim: To Build, change, and destroy AWS / GCP / Microsoft Azure / DigitalOcean infrastructure Using Terraform. (S3 bucket or Docker) fdp.

Terraform: Terraform is an open-source infrastructure as code (IaC) tool that allows you to define, provision, and manage cloud resources across various providers using a declarative configuration language. It enables consistent and repeatable infrastructure deployments, supports multi-cloud environments, and maintains state files to track resource changes. Terraform automates the creation and management of infrastructure, making it easier to scale and modify resources.

Prerequisites:

- Install Terraform .
 - Install Vscode.
 - Hashicorp extension in Vscode.
- AWS Academy Account

Step 1: Open your AWS Academy account . Then Start the lab from modules. After starting the lab click on the AWS details and click on show button after the AWS CLI to get Access keys and other details now copy full credentials.

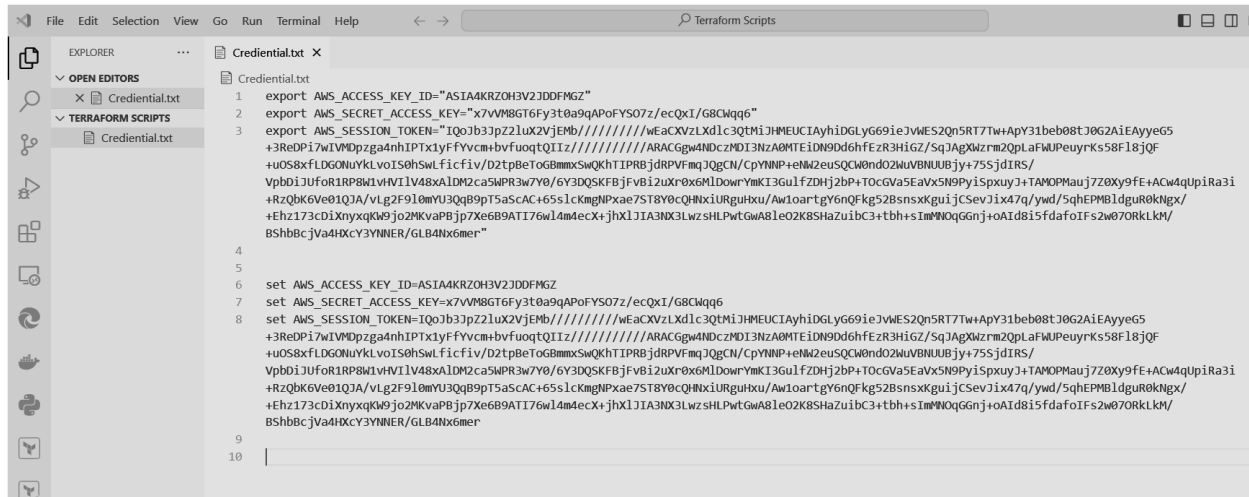


Step 2: Now Create a folder Named As “Terraform Scripts “ and Open it on Vscod or any code editor. Now create one file Credential.txt and paste the copied credentials 2 times .

```
aws_access_key_id=ASIA4KRZOH3V2JDDFMGZ
aws_secret_access_key=x7vVM8GT6Fy3t0a9qAPoFYS07z/ecQxI/G8CWqq6
aws_session_token=IQo7b3jpZ2LuX2VjEMb////////wEaCXVzLXdlc3Q3tMiJHMEUCIAyhiDGLyG69ieJvWES2Qn5RT7Tw+ApY31beb08tJ0G2AIEAyyeG5
+3ReDPi7wIVMDpzga4nhIPTx1yFfYvcm+bfvuoqtQIIz////////ARACGw4NDczMDI3NzA0MTEiDN9Dd6hfEzR3HiGZ/SqJAgXWzrm2QpLaFWUPeuyrKs58F18jQF
+uOS8xFLDGOuYkLvoIS0hSwLficfiv/D2tpBeToGBmmxSwQKhTIPRBjdrPVMfmgJQgCN/CpYNNP+eNM2eUSQCW0nd02WuVBNUUBjy+75SjdIRS/
VpbDiJufor1RP8W1vHV1L48xALDM2ca5WPR3w7Y0/6Y3DQSKFBjFvBi2uXr0x6M1DowrYmKI3Gu1fZDHj2bP+TocGva5EaVx5N9PyiSpxuyJ+TAMOPMauj7Z0Xy9fe+ACw4qUpiRa3i
+RzQbK6Ve0IQJA/vLg2F910mYU3QqB9pT5aScAC+65s1cKmgNPxae7S78Y0cQHnxiURguHxu/Aw1oartgY6nQFkg52BsnxKguijCSevJix47q/ywd/5qhEPMBldguR0kNgx/
+Ehz173cDiXnyxqKw9jo2MKvAPBjp7X66B9ATI76w14m4ecX+jhX1JIA3NX3LwzshLPwtGwa81e02K8SHaZuibC3+tbh+sImMNOqGGnj+oAId8i5fdafoIFs2w070RKLKM/
BShbBcjVa4HxcY3YNNER/GLB4Nx6mer

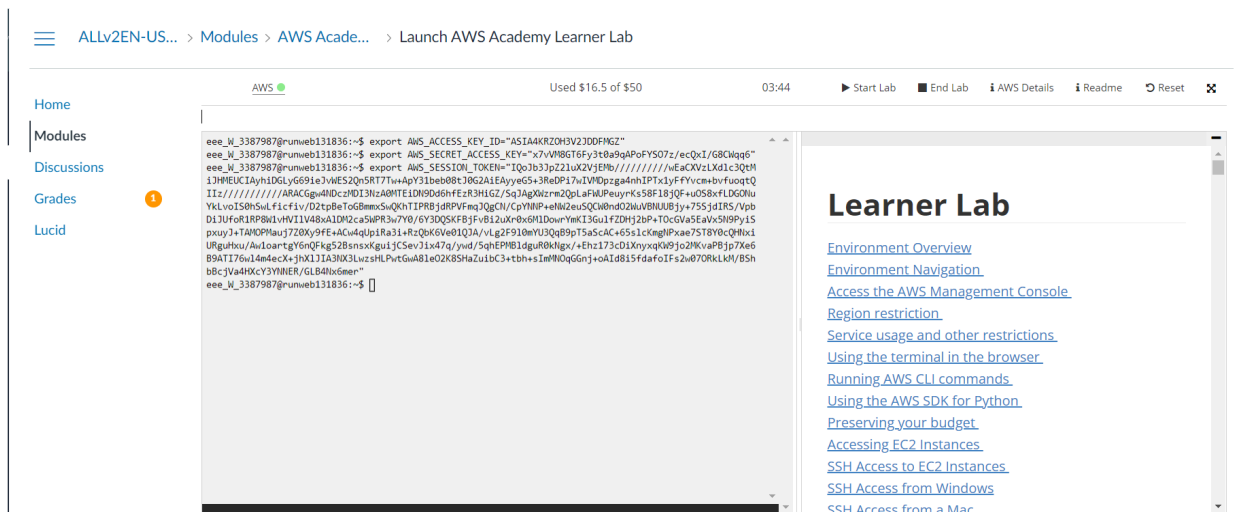
aws_access_key_id=ASIA4KRZOH3V2JDDFMGZ
aws_secret_access_key=x7vVM8GT6Fy3t0a9qAPoFYS07z/ecQxI/G8CWqq6
aws_session_token=IQo7b3jpZ2LuX2VjEMb////////wEaCXVzLXdlc3Q3tMiJHMEUCIAyhiDGLyG69ieJvWES2Qn5RT7Tw+ApY31beb08tJ0G2AIEAyyeG5
+3ReDPi7wIVMDpzga4nhIPTx1yFfYvcm+bfvuoqtQIIz////////ARACGw4NDczMDI3NzA0MTEiDN9Dd6hfEzR3HiGZ/SqJAgXWzrm2QpLaFWUPeuyrKs58F18jQF
+uOS8xFLDGOuYkLvoIS0hSwLficfiv/D2tpBeToGBmmxSwQKhTIPRBjdrPVMfmgJQgCN/CpYNNP+eNM2eUSQCW0nd02WuVBNUUBjy+75SjdIRS/
VpbDiJufor1RP8W1vHV1L48xALDM2ca5WPR3w7Y0/6Y3DQSKFBjFvBi2uXr0x6M1DowrYmKI3Gu1fZDHj2bP+TocGva5EaVx5N9PyiSpxuyJ+TAMOPMauj7Z0Xy9fe+ACw4qUpiRa3i
+RzQbK6Ve0IQJA/vLg2F910mYU3QqB9pT5aScAC+65s1cKmgNPxae7S78Y0cQHnxiURguHxu/Aw1oartgY6nQFkg52BsnxKguijCSevJix47q/ywd/5qhEPMBldguR0kNgx/
+Ehz173cDiXnyxqKw9jo2MKvAPBjp7X66B9ATI76w14m4ecX+jhX1JIA3NX3LwzshLPwtGwa81e02K8SHaZuibC3+tbh+sImMNOqGGnj+oAId8i5fdafoIFs2w070RKLKM/
BShbBcjVa4HxcY3YNNER/GLB4Nx6mer
```

Now Make RHS of both copied keys to upper case and put the LHS of 1st copied keys inside the Double quotes “”
And for 1st copied give prefix export and 2nd copied give prefix set.



```
1 export AWS_ACCESS_KEY_ID="ASIA4KRZOH3V2JDDFMGZ"
2 export AWS_SECRET_ACCESS_KEY="x7vVM8GT6fY3t0a9qAPoFY507z/ecQxI/G8Cwq6"
3 export AWS_SESSION_TOKEN="Iqo3b3pZ21uXZVjEhMb//////////wEaCXVzLXdlc3QtMiJHMEUCIAyhiDGLyG69ieJvMES2Qn5RT7Tw+ApY31beb08tJ0G2AiEAYyeG5
+3RedP17wIWMdpzga4nhIPTx1yFFyVcm+bvfuqQIIz//////////ARACGgw4NDc2MDI3NzA0MTEiDN9DdhFEzR3HiGZ/SqJAgX0zrm2QplAFWUpeuyrKs58F18jQF
+uOS8xfLDG0NuYkLvoIS0hSwLficfiv/D2tpBeToG8mmxSwQKhTIPRBjdRPVfmqJ0gCN/CpYNNP+eNw2eUSQCW0nd02WuVBNUJBjy+755jdIRS/
VpbD1JUfoR1RP8w1vHVI1V48xALDM2ca5WPR3w7Y0/6Y3DQSKFBjfvB12uXr0x6M1DownYmKI3Gu1fZDHj2bP+TocGvA5EavX5N9PyISpxuyJ+TAMOPMauj7Z0Xy9fe+ACw4qUpiRa3i
+RzQbK6ve01QJA/vLg2F910mYU3QqB9pT5aScAC+65s1cKmgNPxae75T8Y0cQHnxiURguHbu/Aw1oartgY6nQFkg52BsnxkguijCsev3ix47q/ywd/5qhEPMB1dguR0kNgx/
+Ehz173cDIxnyxqKw9jo2MKvAPbjp7X6B9ATI76w14m4ecX+jhX1JIA3NX3LwzSHLPwtGwA81e02K8SHAzuibC3+tbh+sImPwQ6Ggnj+oAI815fdafoIFs2w070RkLkM/
BSHbhcjVa4HxcY3YNNER/GLB4Nx6mer"
4
5
6 set AWS_ACCESS_KEY_ID=ASIA4KRZOH3V2JDDFMGZ
7 set AWS_SECRET_ACCESS_KEY=x7vVM8GT6fY3t0a9qAPoFY507z/ecQxI/G8Cwq6
8 set AWS_SESSION_TOKEN=Iqo3b3pZ21uXZVjEhMb//////////wEaCXVzLXdlc3QtMiJHMEUCIAyhiDGLyG69ieJvMES2Qn5RT7Tw+ApY31beb08tJ0G2AiEAYyeG5
+3RedP17wIWMdpzga4nhIPTx1yFFyVcm+bvfuqQIIz//////////ARACGgw4NDc2MDI3NzA0MTEiDN9DdhFEzR3HiGZ/SqJAgX0zrm2QplAFWUpeuyrKs58F18jQF
+uOS8xfLDG0NuYkLvoIS0hSwLficfiv/D2tpBeToG8mmxSwQKhTIPRBjdRPVfmqJ0gCN/CpYNNP+eNw2eUSQCW0nd02WuVBNUJBjy+755jdIRS/
VpbD1JUfoR1RP8w1vHVI1V48xALDM2ca5WPR3w7Y0/6Y3DQSKFBjfvB12uXr0x6M1DownYmKI3Gu1fZDHj2bP+TocGvA5EavX5N9PyISpxuyJ+TAMOPMauj7Z0Xy9fe+ACw4qUpiRa3i
+RzQbK6ve01QJA/vLg2F910mYU3QqB9pT5aScAC+65s1cKmgNPxae75T8Y0cQHnxiURguHbu/Aw1oartgY6nQFkg52BsnxkguijCsev3ix47q/ywd/5qhEPMB1dguR0kNgx/
+Ehz173cDIxnyxqKw9jo2MKvAPbjp7X6B9ATI76w14m4ecX+jhX1JIA3NX3LwzSHLPwtGwA81e02K8SHAzuibC3+tbh+sImPwQ6Ggnj+oAI815fdafoIFs2w070RkLkM/
BSHbhcjVa4HxcY3YNNER/GLB4Nx6mer
9
10
```

Step 3: Copy paste the 1st paragraph or 1st copied in learners lab terminal and hit enter.



Home Modules Discussions Grades Lucid

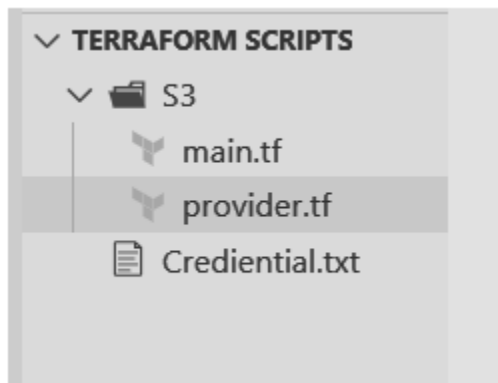
AWS Use \$16.5 of \$50 03:44 Start Lab End Lab AWS Details Readme Reset

```
eee_M_3387987@runneb131836:~$ export AWS_ACCESS_KEY_ID="ASIA4KRZOH3V2JDDFMGZ"
eee_M_3387987@runneb131836:~$ export AWS_SECRET_ACCESS_KEY="x7vVM8GT6fY3t0a9qAPoFY507z/ecQxI/G8Cwq6"
eee_M_3387987@runneb131836:~$ export AWS_SESSION_TOKEN="Iqo3b3pZ21uXZVjEhMb//////////wEaCXVzLXdlc3QtMiJHMEUCIAyhiDGLyG69ieJvMES2Qn5RT7Tw+ApY31beb08tJ0G2AiEAYyeG5
+3RedP17wIWMdpzga4nhIPTx1yFFyVcm+bvfuqQIIz//////////ARACGgw4NDc2MDI3NzA0MTEiDN9DdhFEzR3HiGZ/SqJAgX0zrm2QplAFWUpeuyrKs58F18jQF
+uOS8xfLDG0NuYkLvoIS0hSwLficfiv/D2tpBeToG8mmxSwQKhTIPRBjdRPVfmqJ0gCN/CpYNNP+eNw2eUSQCW0nd02WuVBNUJBjy+755jdIRS/
VpbD1JUfoR1RP8w1vHVI1V48xALDM2ca5WPR3w7Y0/6Y3DQSKFBjfvB12uXr0x6M1DownYmKI3Gu1fZDHj2bP+TocGvA5EavX5N9PyISpxuyJ+TAMOPMauj7Z0Xy9fe+ACw4qUpiRa3i
+RzQbK6ve01QJA/vLg2F910mYU3QqB9pT5aScAC+65s1cKmgNPxae75T8Y0cQHnxiURguHbu/Aw1oartgY6nQFkg52BsnxkguijCsev3ix47q/ywd/5qhEPMB1dguR0kNgx/
+Ehz173cDIxnyxqKw9jo2MKvAPbjp7X6B9ATI76w14m4ecX+jhX1JIA3NX3LwzSHLPwtGwA81e02K8SHAzuibC3+tbh+sImPwQ6Ggnj+oAI815fdafoIFs2w070RkLkM/
BSHbhcjVa4HxcY3YNNER/GLB4Nx6mer"
eee_M_3387987@runneb131836:~$
```

Learner Lab

- [Environment Overview](#)
- [Environment Navigation](#)
- [Access the AWS Management Console](#)
- [Region restriction](#)
- [Service usage and other restrictions](#)
- [Using the terminal in the browser](#)
- [Running AWS CLI commands](#)
- [Using the AWS SDK for Python](#)
- [Preserving your budget](#)
- [Accessing EC2 Instances](#)
- [SSH Access to EC2 Instances](#)
- [SSH Access from Windows](#)
- [SSH Access from a Mac](#)

Step 4: Now create S3 folder inside the same folder and create provider.tf and main.tf file in S3 folder.



Step 5: In provider.tf add the following script and do not forget to add your key values and region inside the " " quotes all details you will get in AWS details in lab .

```
provider "aws" {
  access_key=""
  secret_key=""
  token=""
  region=""
}
```

```
provider "aws" {
  access_key=""
  secret_key=""
  token=""
  region=""
}
```

Step 6 : Visit the website Terraform By Hashicorp go inside registry then click on Browse provider then click on or search AWS then click on use provider and copy the code exact provider part. Because provider part we have already cover. Paste the code in main.tf.

The screenshot shows the Terraform AWS provider page. The main content area displays the AWS logo, the provider name 'aws', and its version '5.64.0'. It also shows the publisher 'HashiCorp' and the release date '2 days ago'. A description states: 'Lifecycle management of AWS resources, including EC2, Lambda, EKS, ECS, VPC, S3, RDS, DynamoDB, and more. This provider is maintained internally by the HashiCorp AWS Provider team.' On the right side, there is a section titled 'How to use this provider' which provides instructions on how to install the provider and a code snippet for the Terraform configuration.

How to use this provider

To install this provider, copy and paste this code into your Terraform configuration. Then, run terraform init.

Terraform 0.13+

```
terraform {
  required_providers {
    aws = {
      source = "hashicorp/aws"
      version = "5.64.0"
    }
  }
}
```

provider "aws" {

Configuration options

```
terraform {  
  required_providers {  
    aws = {  
      source = "hashicorp/aws"  
      version = "5.64.0"  
    }  
  }  
}
```



Step 7: Now write the following code in main.tf to create the bucket.

```
resource "aws_s3_bucket" "any_name" {  
  
  bucket = "Bucket_Name_It_sholud_be_Unique_and_all_in_lowercase_follow_Documentation"  
  tags = {  
    Name = "Any_name"  
  }  
}
```

```
Credential.txt  provider.tf  main.tf  X
S3 > main.tf > resource "aws_s3_bucket" "my-bucket-us-east-1" > bucket
1  terraform {
2      required_providers {
3          aws = {
4              source = "hashicorp/aws"
5              version = "5.64.0"
6          }
7      }
8  }
9
10 resource "aws_s3_bucket" "my-bucket-us-east-1" {
11
12     bucket = "bhushan-kor-aws-bucket-terraform"
13     tags = {
14         Name = "Sample Bucket"
15     }
16 }
```

Step 8: Now open new terminal and cd to S3 And perform the following commands

- 1) terraform init
- 2) terraform plan
- 3) terraform apply

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\Users\bhush\one drive 2\OneDrive\Desktop\Terraform Scripts>cd S3
```

```
C:\Users\bhush\one drive 2\OneDrive\Desktop\Terraform Scripts\S3>terraform init
Initializing the backend...
```

```
Initializing provider plugins...
```

- Finding hashicorp/aws versions matching "5.64.0"...
- Installing hashicorp/aws v5.64.0...
- Installed hashicorp/aws v5.64.0 (signed by HashiCorp)

Terraform has created a lock file **.terraform.lock.hcl** to record the provider selections it made above. Include this file in your version control repository so that Terraform can guarantee to make the same selections by default when you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see any changes that are required for your infrastructure. All Terraform commands should now work.

If you ever set or change modules or backend configuration for Terraform, rerun this command to reinitialize your working directory. If you forget, other commands will detect it and remind you to do so if necessary.

```
C:\Users\bhush\one drive 2\OneDrive\Desktop\Terraform Scripts\S3>terraform plan
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

```
# aws_s3_bucket.my-bucket-us-east-1 will be created
+ resource "aws_s3_bucket" "my-bucket-us-east-1" {
  + acceleration_status = (known after apply)
  + acl                 = (known after apply)
  + arn                 = (known after apply)
  + bucket              = "bhushan-Kor-aws-bucket-terraform"
  + bucket_domain_name = (known after apply)
  + bucket_prefix       = (known after apply)
  + bucket_regional_domain_name = (known after apply)
  + force_destroy       = false
  + hosted_zone_id      = (known after apply)
  + id                  = (known after apply)
  + object_lock_enabled = (known after apply)
  + policy              = (known after apply)
  + region              = (known after apply)
  + request_payer       = (known after apply)
  + tags                = {
    + "Name" = "Sample Bucket"
  }
  + tags_all            = {
    + "Name" = "Sample Bucket"
  }
  + website_domain      = (known after apply)
  + website_endpoint    = (known after apply)

  + cors_rule (known after apply)

  + grant (known after apply)

  + lifecycle_rule (known after apply)
```

```
+ lifecycle_rule (known after apply)
+ logging (known after apply)
+ object_lock_configuration (known after apply)
+ replication_configuration (known after apply)
+ server_side_encryption_configuration (known after apply)
+ versioning (known after apply)
+ website (known after apply)
}
```

Plan: 1 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.

C:\Users\bhush\OneDrive\OneDrive\Desktop\Terraform Scripts\S3>terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

```
# aws_s3_bucket.my-bucket-us-east-1 will be created
+ resource "aws_s3_bucket" "my-bucket-us-east-1" {
  + acceleration_status = (known after apply)
  + acl                 = (known after apply)
  + arn                 = (known after apply)
  + bucket              = "bhushan-kor-aws-bucket-terraform"
  + bucket_domain_name = (known after apply)
  + bucket_prefix       = (known after apply)
  + bucket_regional_domain_name = (known after apply)
  + force_destroy       = false
  + hosted_zone_id      = (known after apply)
  + id                  = (known after apply)
  + object_lock_enabled = (known after apply)
  + policy              = (known after apply)
  + region              = (known after apply)
  + request_payer       = (known after apply)
  + tags                = {
    + "Name" = "Sample Bucket"
  }
  + tags_all            = {
    + "Name" = "Sample Bucket"
  }
  + website_domain      = (known after apply)
  + website_endpoint    = (known after apply)

  + cors_rule (known after apply)
  + grant (known after apply)
```



```
+ cors_rule (known after apply)
+ grant (known after apply)
+ lifecycle_rule (known after apply)
+ logging (known after apply)
+ object_lock_configuration (known after apply)
+ replication_configuration (known after apply)
+ server_side_encryption_configuration (known after apply)
+ versioning (known after apply)
+ website (known after apply)
+ website (known after apply)
}
+ website (known after apply)
}
```

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_s3_bucket.my-bucket-us-east-1: Creating...

aws_s3_bucket.my-bucket-us-east-1: Creation complete after 8s [id=bhushan-kor-aws-bucket-terraform]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

```
+ website (known after apply)
}
```

```
+ website (known after apply)
}
```

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_s3_bucket.my-bucket-us-east-1: Creating...

aws_s3_bucket.my-bucket-us-east-1: Creation complete after 8s [id=bhushan-kor-aws-bucket-terraform]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

```
C:\Users\bhush\one drive 2\OneDrive\Desktop\Terraform Scripts\S3>
+ website (known after apply)
}
```

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_s3_bucket.my-bucket-us-east-1: Creating...

aws_s3_bucket.my-bucket-us-east-1: Creation complete after 8s [id=bhushan-kor-aws-bucket-terraform]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

```
C:\Users\bhush\one drive 2\OneDrive\Desktop\Terraform Scripts\S3>
```

```
+ website (known after apply)
```

```
+ website (known after apply)
}
```

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_s3_bucket.my-bucket-us-east-1: Creating...

aws_s3_bucket.my-bucket-us-east-1: Creation complete after 8s [id=bhushan-kor-aws-bucket-terraform]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

C:\Users\bhush\one drive 2\OneDrive\Desktop\Terraform Scripts\S3>

```
}
```

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_s3_bucket.my-bucket-us-east-1: Creating...

aws_s3_bucket.my-bucket-us-east-1: Creation complete after 8s [id=bhushan-kor-aws-bucket-terraform]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

C:\Users\bhush\one drive 2\OneDrive\Desktop\Terraform Scripts\S3>

Now If you Click on AWS on lab it will open your AWS account and in S3 you can see you bucket.
Before terrafrom Apply:

The screenshot shows the Amazon S3 console interface. On the left is a navigation menu with options like Buckets, Access Grants, and Storage Lens. The main area displays 'General purpose buckets (5)' with a table listing existing buckets. The table has columns for Name, AWS Region, IAM Access Analyzer, and Creation date. The buckets listed are bhushan2, elasticbeanstalk-us-east-1-847302770411, exp6bucket, s3bucketexp6, and staticwebhosting28, all in the US East (N. Virginia) us-east-1 region.

Name	AWS Region	IAM Access Analyzer	Creation date
bhushan2	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 20:17:09 (UTC+05:30)
elasticbeanstalk-us-east-1-847302770411	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 5, 2024, 14:00:50 (UTC+05:30)
exp6bucket	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 19:57:25 (UTC+05:30)
s3bucketexp6	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 20:08:51 (UTC+05:30)
staticwebhosting28	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 4, 2024, 23:33:54 (UTC+05:30)

Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

Roll No: 28

After terraform apply:

Account snapshot - updated every 24 hours [All AWS Regions](#) [View Storage Lens dashboard](#)

Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

General purpose buckets Directory buckets

General purpose buckets (6) [Info](#) [All AWS Regions](#)

Buckets are containers for data stored in S3.

Name	AWS Region	IAM Access Analyzer	Creation date
bhushan-kor-aws-bucket-terraform	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 25, 2024, 12:12:05 (UTC+05:30)
bhushan2	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 20:17:09 (UTC+05:30)
elasticbeanstalk-us-east-1-847302770411	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 5, 2024, 14:00:50 (UTC+05:30)
exp6bucket	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 19:57:25 (UTC+05:30)
s3bucketexp6	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 20:08:51 (UTC+05:30)
staticwebhosting28	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 4, 2024, 23:33:54 (UTC+05:30)

Step 9: Now to destroy the bucket run command terraform destroy.

```
C:\Users\bhush\one drive 2\OneDrive\Desktop\Terraform Scripts\S3>terraform destroy
aws_s3_bucket.my-bucket-us-east-1: Refreshing state... [id=bhushan-kor-aws-bucket-terraform]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
- destroy

Terraform will perform the following actions:

# aws_s3_bucket.my-bucket-us-east-1 will be destroyed
- resource "aws_s3_bucket" "my-bucket-us-east-1" {
  - arn                = "arn:aws:s3:::bhushan-kor-aws-bucket-terraform" -> null
  - bucket             = "bhushan-kor-aws-bucket-terraform" -> null
  - bucket_domain_name = "bhushan-kor-aws-bucket-terraform.s3.amazonaws.com" -> null
  - bucket_regional_domain_name = "bhushan-kor-aws-bucket-terraform.s3.us-east-1.amazonaws.com" -> null
  - force_destroy      = false -> null
  - hosted_zone_id     = "Z3AQBSTGFYJSTF" -> null
  - id                 = "bhushan-kor-aws-bucket-terraform" -> null
  - object_lock_enabled = false -> null
  - region             = "us-east-1" -> null
  - request_payer      = "BucketOwner" -> null
  - tags               = {
    - "Name" = "Sample Bucket"
  } -> null
  - tags_all           = {
    - "Name" = "Sample Bucket"
  } -> null
  # (3 unchanged attributes hidden)

- grant {
  - id          = "778043bd0e67860760caebd7a1a61d745d8798fa35ab31144e54d7003ee08ae8" -> null
  - permissions = [
    - "FULL_CONTROL",
  ] -> null
  - type        = "CanonicalUser" -> null
  # (1 unchanged attribute hidden)
}
```

```
- "FULL_CONTROL",
] -> null
- type      = "CanonicalUser" -> null
  # (1 unchanged attribute hidden)
}

- server_side_encryption_configuration {
  - rule {
    - bucket_key_enabled = false -> null

    - apply_server_side_encryption_by_default {
      - sse_algorithm = "AES256" -> null
        # (1 unchanged attribute hidden)
    }
  }
}

- versioning {
  - enabled = false -> null
  - mfa_delete = false -> null
}
```

Plan: 0 to add, 0 to change, 1 to destroy.

Do you really want to destroy all resources?

Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

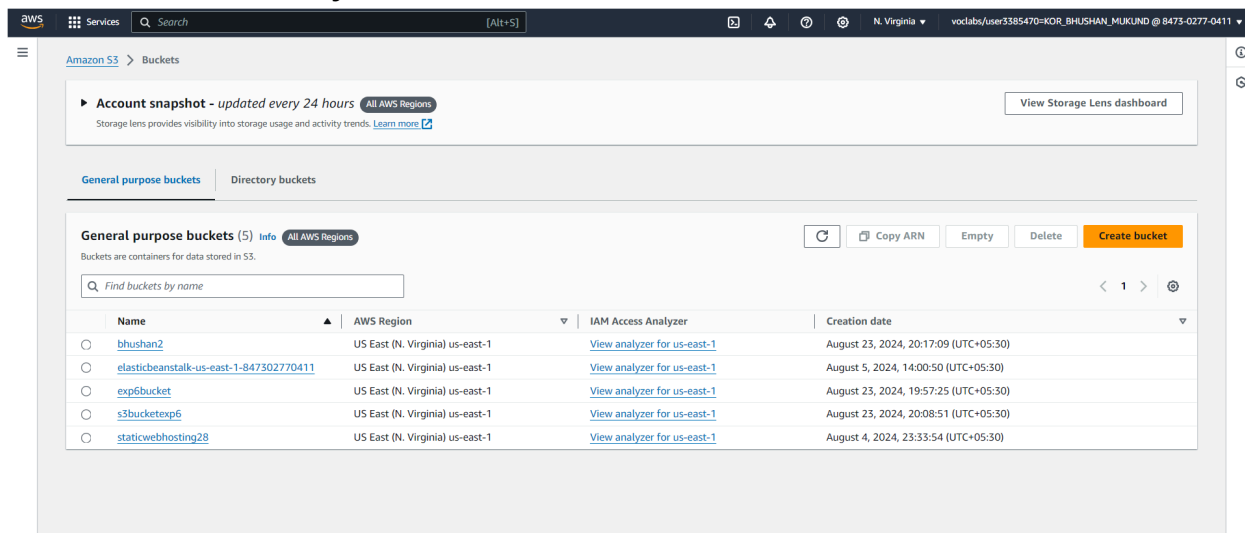
aws_s3_bucket.my-bucket-us-east-1: Destroying... [id=bhushan-kor-aws-bucket-terraform]

aws_s3_bucket.my-bucket-us-east-1: Destruction complete after 2s

Destroy complete! Resources: 1 destroyed.

C:\Users\bhush\one drive 2\OneDrive\Desktop\Terraform Scripts\S3>

After terraform destroy :



Account snapshot - updated every 24 hours [All AWS Regions](#) [View Storage Lens dashboard](#)

Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

General purpose buckets | Directory buckets

General purpose buckets (5) [Info](#) [All AWS Regions](#)

Buckets are containers for data stored in S3.

Find buckets by name

Name	AWS Region	IAM Access Analyzer	Creation date
bhushan2	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 20:17:09 (UTC+05:30)
elasticbeanstalk-us-east-1-847302770411	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 5, 2024, 14:00:50 (UTC+05:30)
exp6bucket	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 19:57:25 (UTC+05:30)
s3bucketexp6	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 20:08:51 (UTC+05:30)
staticwebhosting28	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 4, 2024, 23:33:54 (UTC+05:30)

Step 10: Congratulations we are done with creating and destroying the S3 bucket on AWS using Terraform.