s3.tf

```
resource "aws_s3_bucket" "jyoti" {
  bucket = "unique-jyoti-bucket-2024"

  tags = {
    Name        = "My Bucket"
    Environment = "Dev"
  }
}
```

```
provider "aws" {
    access_key="ASIAX5FJEHDXR2O2T2G6"
    secret_key="Gsbh6rsw6nvLWmL1CUkmT4I6MS+p0osKEPJWpsfA"
    token="IQoJb3JpZ2luX2VjEJf//////////wEaCXVzLXdlc3QtMi
    region="us-east-1"
}
```

```
PS C:\terraform_script> cd C:\terraform_script\s3
```

```
S C:\terraform_script\s3> terraform init
nitializing the backend...
nitializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.64.0

erraform has been successfully initialized!

ou may now begin working with Terraform. Try running "terraform plan" to
see
ny changes that are required for your infrastructure. All Terraform comm
nds
hould now work.

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

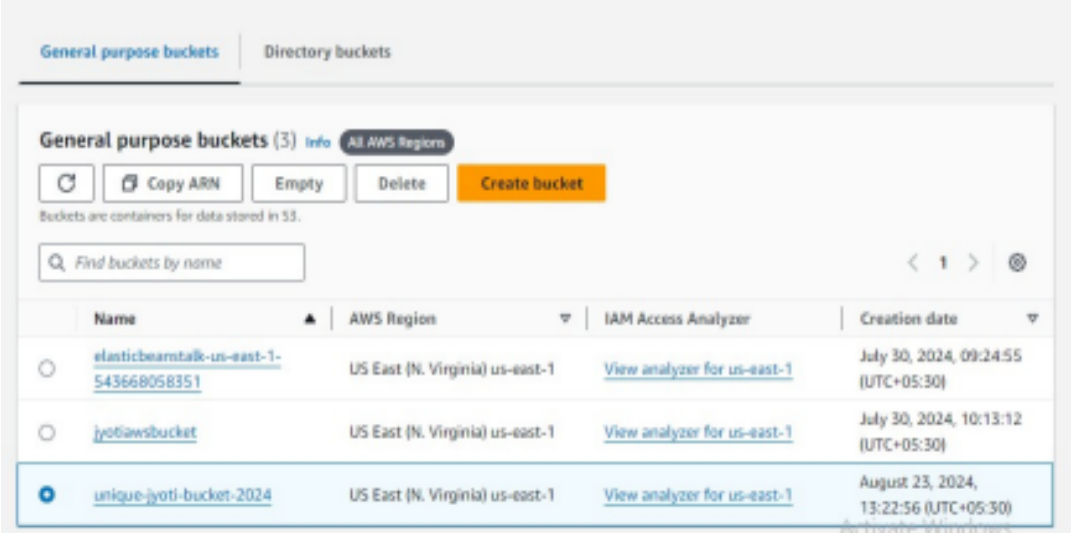
```
PS C:\terraform_script\s3> terraform plan

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.jyoti will be created
  + resource "aws_s3_bucket" "jyoti" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "unique-jyoti-bucket-2024"
      + 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                        = {
          + "Environment" = "Dev"
          + "Name"        = "My Bucket"
        }
```

```
      }
    + tags_all                    = {
        + "Environment" = "Dev"
        + "Name"        = "My 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)

    + 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.
```

```
PS C:\terraform_script\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.jyoti will be created
  + resource "aws_s3_bucket" "jyoti" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "unique-jyoti-bucket-2024"
      + 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                        = {
          + "Environment" = "Dev"
          + "Name"        = "My Bucket"
```

```
    }
  + tags_all                    = {
      + "Environment" = "Dev"
      + "Name"        = "My 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)


  + logging (known after apply)


  + object_lock_configuration (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)
  + versioning (known after apply)
```

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

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

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

Do you want to perform these actions?
  Terraform will perform the actions described above.
Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.
  Only 'yes' will be accepted to approve.


  Enter a value: yes


aws_s3_bucket.jyoti: Creating...
aws_s3_bucket.jyoti: Creating...
aws_s3_bucket.jyoti: Creation complete after 6s [id=unique-jyoti-bucket-2aws_s3_buc
ti-bucket-2024]


Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
PS C:\terraform_script\s3>
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```



```
PS C:\terraform_script\s3> terraform destroy
aws_s3_bucket.jyoti: Refreshing state... [id=unique-jyoti-bucket-2024]

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

Terraform will perform the following actions:

  # aws_s3_bucket.jyoti will be destroyed
  - resource "aws_s3_bucket" "jyoti" {
      - arn                         = "arn:aws:s3:::unique-jyoti-bucket-2024" -> null
      - bucket                      = "unique-jyoti-bucket-2024" -> null
      - bucket_domain_name          = "unique-jyoti-bucket-2024.s3.amazonaws.com" -> null
      - bucket_regional_domain_name = "unique-jyoti-bucket-2024.s3.us-east-1.amazonaws.com" -> null
      - force_destroy               = false -> null
      - hosted_zone_id              = "Z3AQBSTGFYJSTF" -> null
      - id                          = "unique-jyoti-bucket-2024" -> null
      - object_lock_enabled         = false -> null
      - region                      = "us-east-1" -> null
      - request_payer               = "BucketOwner" -> null
      - tags                        = {
          - "Environment" = "Dev"
          - "Name"        = "My Bucket"
        } -> null
      - tags_all                    = {
          - "Environment" = "Dev"
          - "Name"        = "My Bucket"
        } -> 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.jyoti: Destroying... [id=unique-jyoti-bucket-2024]
  aws_s3_bucket.jyoti: Destruction complete after 1s

  Destroy complete! Resources: 1 destroyed.
  PS C:\terraform_script\s3> terraform destroy

  No changes. No objects need to be destroyed.

  Either you have not created any objects yet or the existing objects were already deleted outside

  Destroy complete! Resources: 0 destroyed.
```