ASSIGNMENT 5.5B

on

Infrastructure as code

Submitted by:

Haseebullah Shaikh (2303.KHI.DEG.015)

Faiza Gulzar Ahmed (2303.khi.deg.001)

and

Aqsa Tauheed (2303.khi.deg.011)

Dated: 22nd May 2023

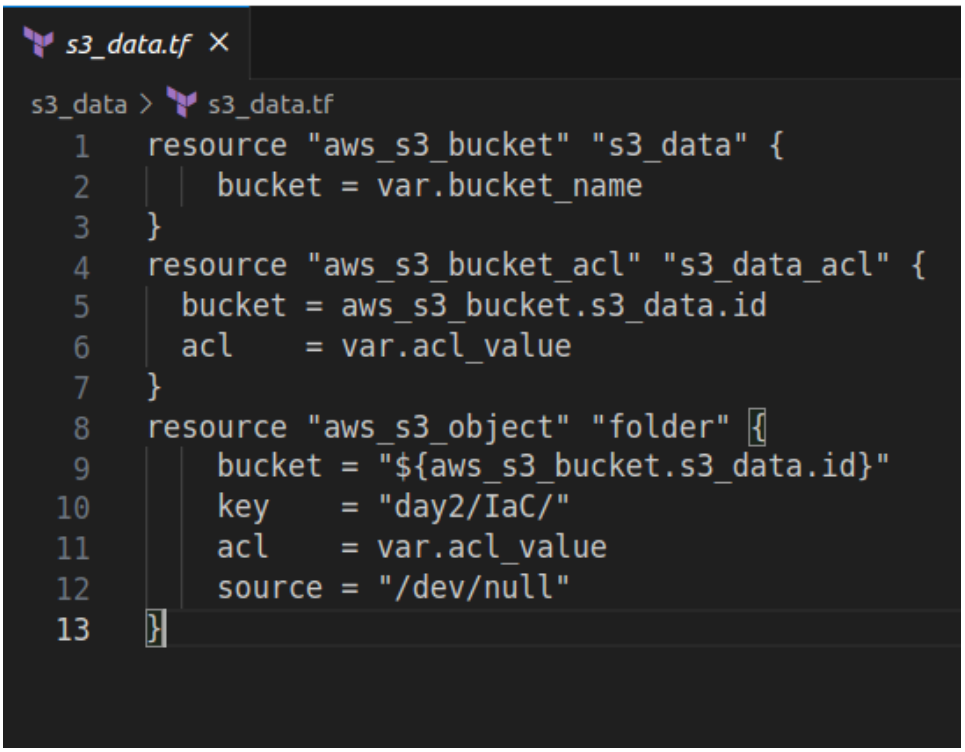
Solution:

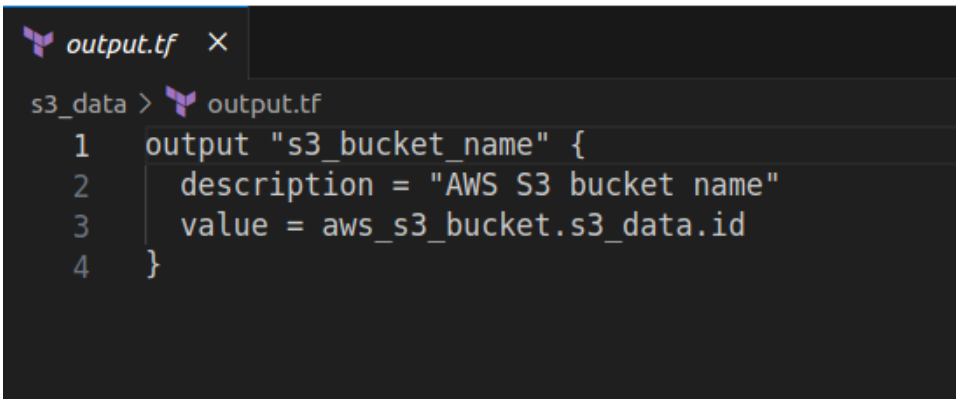
Created files for creating module S3_data.tf, variables.tf, output.tf.

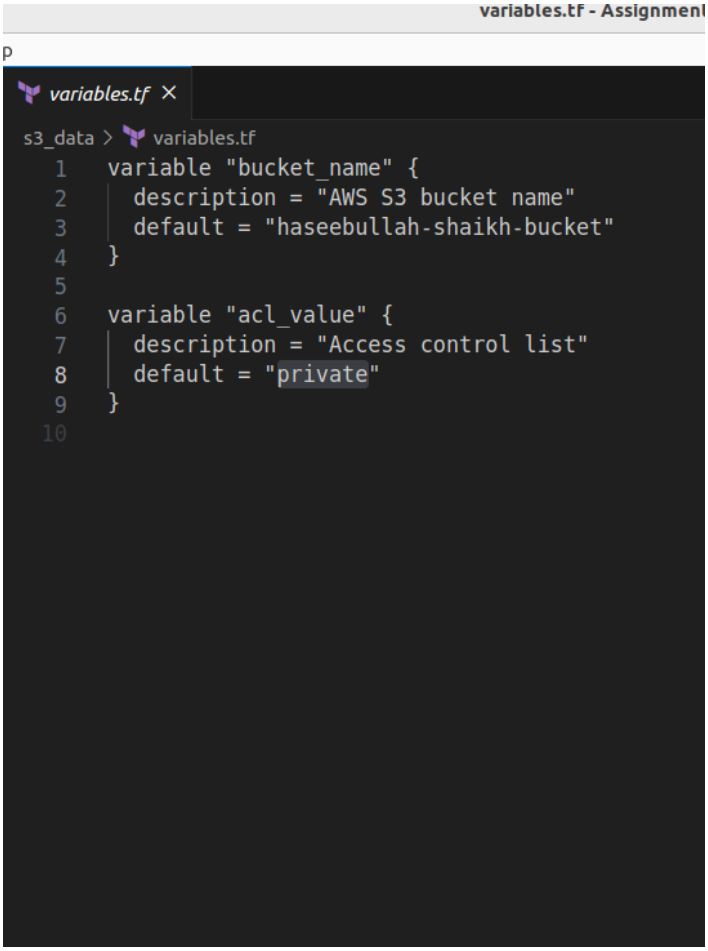
Created three resources in module file.

1. To create bucket named: Haseebullah-shaikh-bucket
2. To set acl value, for making it to private use only
3. To add give folder directory “day2IaC/

Variables that are assigned in following s3_data.tf are defined in variables.tf also output file is created separately, which will be access in main.tf where this module will be called as source.






variables.tf - Assignment
p
variables.tf
s3_data > variables.tf
variable "bucket_name" {
description = "AWS S3 bucket name"
default = "haseebullah-shaikh-bucket"
}
variable "acl_value" {
description = "Access control list"
default = "private"
}

Main.tf to access created module, the module is accesed through file path ./s3_data,

Variables that are assigned are defined in separate file named variables.tf

main.tf

```
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = ">= 3.20.0"
    }
  }
}

provider "aws" {
  region  = "us-east-1"
}

module "s3_data" {
  source      = "./s3_data"
  bucket_name = var.bucket_name
}

output "s3_bucket_id" {
  value = module.s3_data.s3_bucket_name
}
```

variables.tf

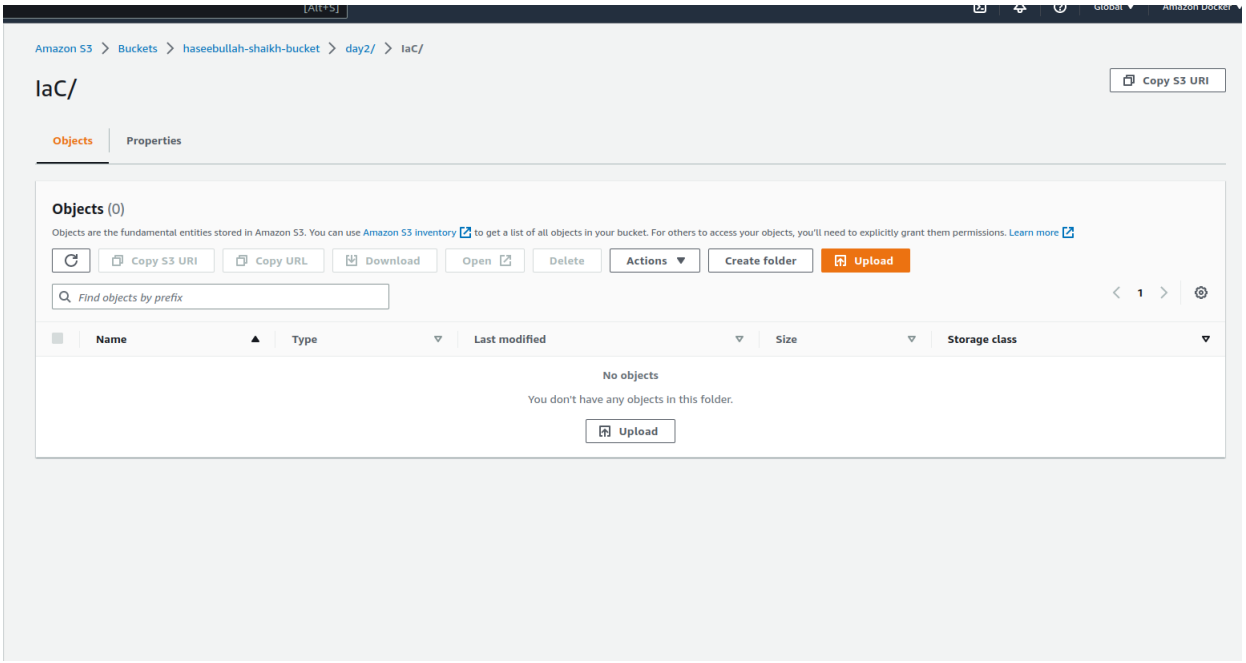
```
# S3
variable "bucket_name" {
  description = "AWS S3 bucket name"
  default = "haseebullah-shaikh-bucket"
}
```

Commands to create the bucket

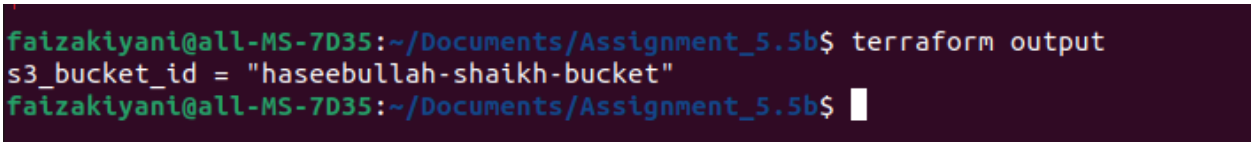
```
remove_bucket: haseebullah-shaikh-bucket
faizakiyani@all-MS-7D35:~/Documents/Assignment_5.5b$ terraform init

Initializing the backend...
Initializing modules...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v4.67.0

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.
faizakiyani@all-MS-7D35:~/Documents/Assignment_5.5b$ terraform apply
```

```
faizakiyani@all-MS-7D35:~/Documents/Assignment_5.5b$ 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:

  # module.s3_data.aws_s3_bucket.s3_data will be created
  + resource "aws_s3_bucket" "s3_data" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "haseebullah-shaikh-bucket"
      + 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_all                    = (known after apply)
      + website_domain              = (known after apply)
      + website_endpoint            = (known after apply)
    }

  # module.s3_data.aws_s3_bucket_acl.s3_data_acl will be created
  + resource "aws_s3_bucket_acl" "s3_data_acl" {
      + acl    = "private"
      + bucket = (known after apply)
      + id     = (known after apply)
    }

  # module.s3_data.aws_s3_object.folder will be created
  + resource "aws_s3_object" "folder" {
      + acl                    = "private"
      + bucket                 = (known after apply)
      + bucket_key_enabled     = (known after apply)
      + content_type           = (known after apply)
      + etag                   = (known after apply)
      + force_destroy          = false
      + id                     = (known after apply)
      + key                    = "day2/IaC/"
      + kms_key_id             = (known after apply)
      + server_side_encryption = (known after apply)
      + source                 = "/dev/null"
      + storage_class          = (known after apply)
      + tags_all               = (known after apply)
      + version_id             = (known after apply)
    }

Plan: 3 to add, 0 to change, 0 to destroy.

Changes to Outputs:
```

Verifying bucket is created with given folder directory.



Command to print bucket it to the output



Command to destroy the bucket

```
s3_bucket_id = "haseebullah-shaikh-bucket"
faizakiyani@all-MS-7D35:~/Documents/Assignment_5.5b$ terraform destroy
module.s3_data.aws_s3_bucket.s3_data: Refreshing state... [id=haseebullah-shaikh-bucket]
module.s3_data.aws_s3_object.folder: Refreshing state... [id=day2/IaC/]

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:

  # module.s3_data.aws_s3_bucket.s3_data will be destroyed
  - resource "aws_s3_bucket" "s3_data" {
      - arn                         = "arn:aws:s3:::haseebullah-shaikh-bucket" -> null
      - bucket                      = "haseebullah-shaikh-bucket" -> null
      - bucket_domain_name          = "haseebullah-shaikh-bucket.s3.amazonaws.com" -> null
      - bucket_regional_domain_name = "haseebullah-shaikh-bucket.s3.amazonaws.com" -> null
      - force_destroy               = false -> null
      - hosted_zone_id              = "Z3AQBSTGFYJSTF" -> null
      - id                          = "haseebullah-shaikh-bucket" -> null
      - object_lock_enabled         = false -> null
      - region                      = "us-east-1" -> null
      - request_payer               = "BucketOwner" -> null
    }

Plan: 0 to add, 0 to change, 2 to destroy.

Changes to Outputs:
  - s3_bucket_id = "haseebullah-shaikh-bucket" -> null

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

module.s3_data.aws_s3_object.folder: Destroying... [id=day2/IaC/]
module.s3_data.aws_s3_object.folder: Destruction complete after 1s
module.s3_data.aws_s3_bucket.s3_data: Destroying... [id=haseebullah-shaikh-buc
module.s3_data.aws_s3_bucket.s3_data: Destruction complete after 1s

Destroy complete! Resources: 2 destroyed.
faizakiyani@all-MS-7D35:~/Documents/Assignment_5.5b$
```

The End ☺