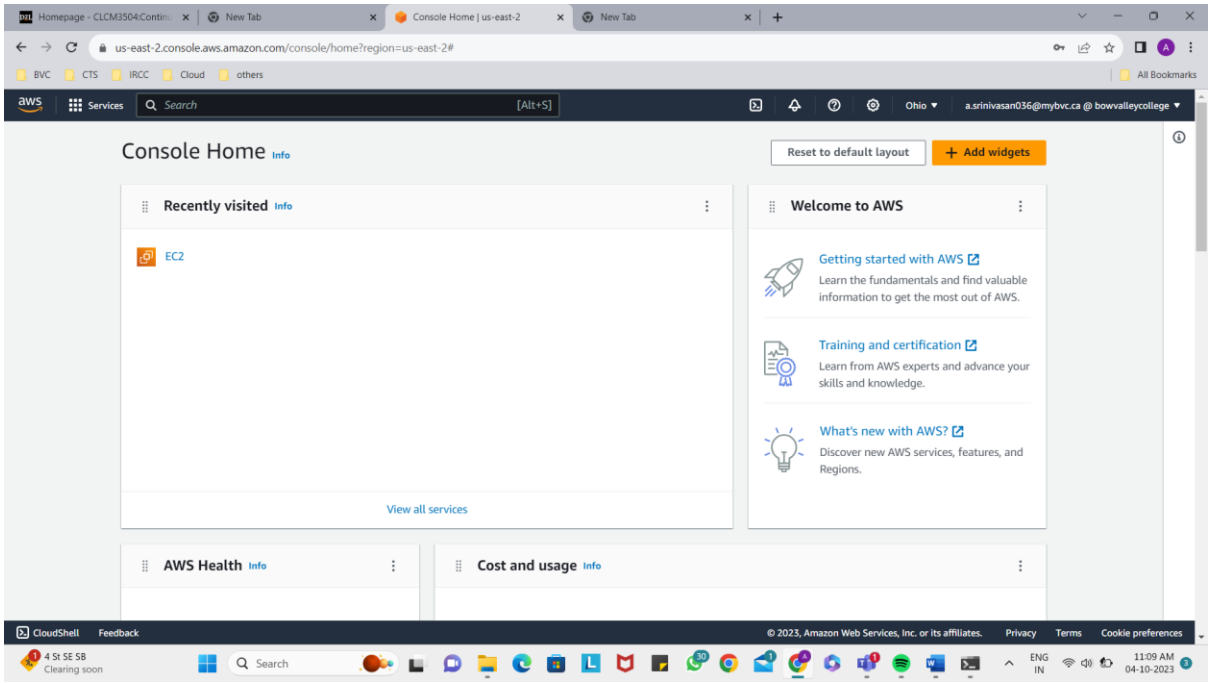
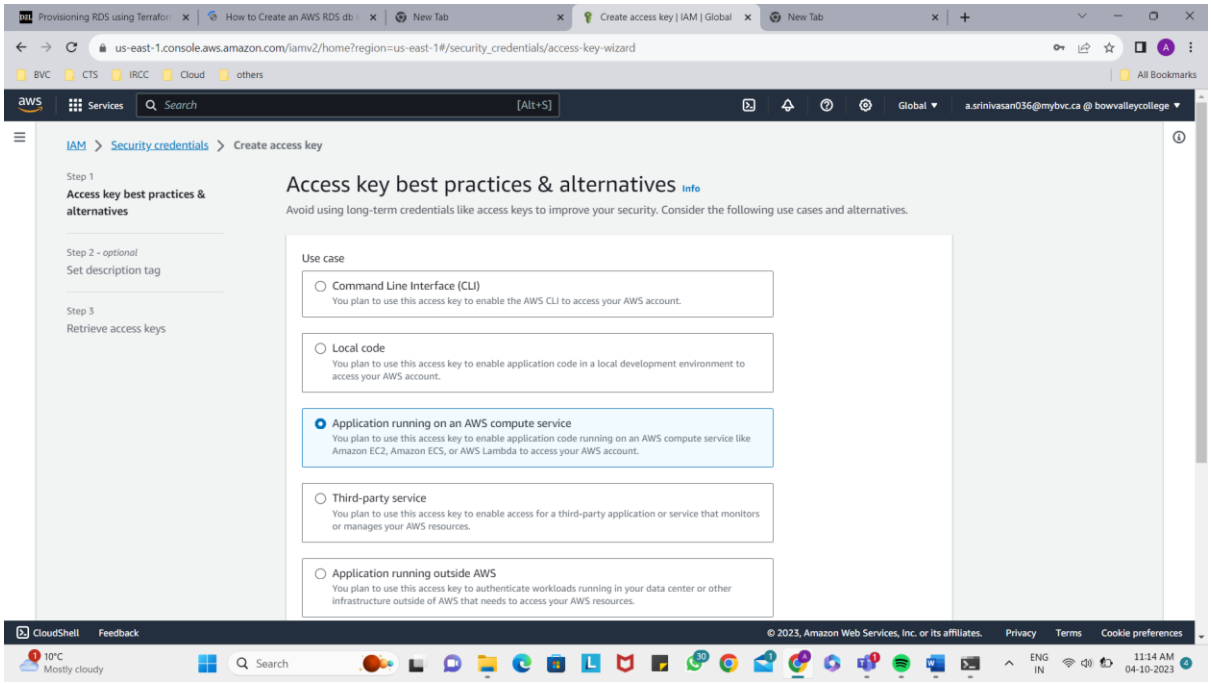
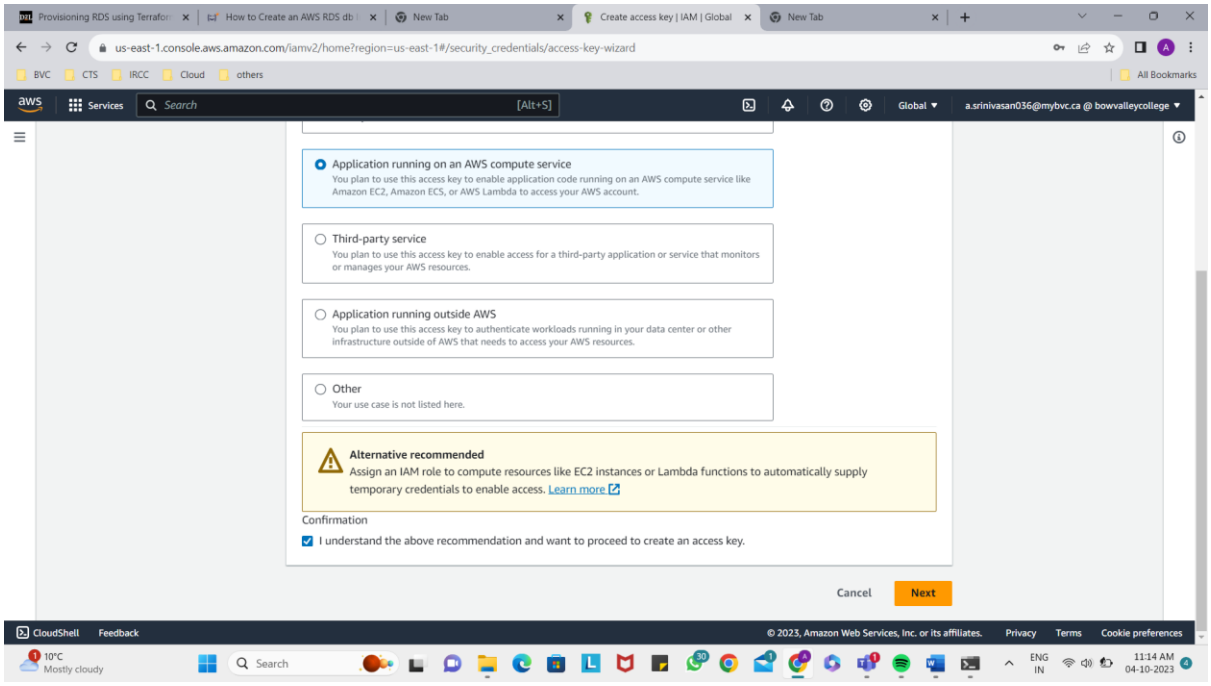
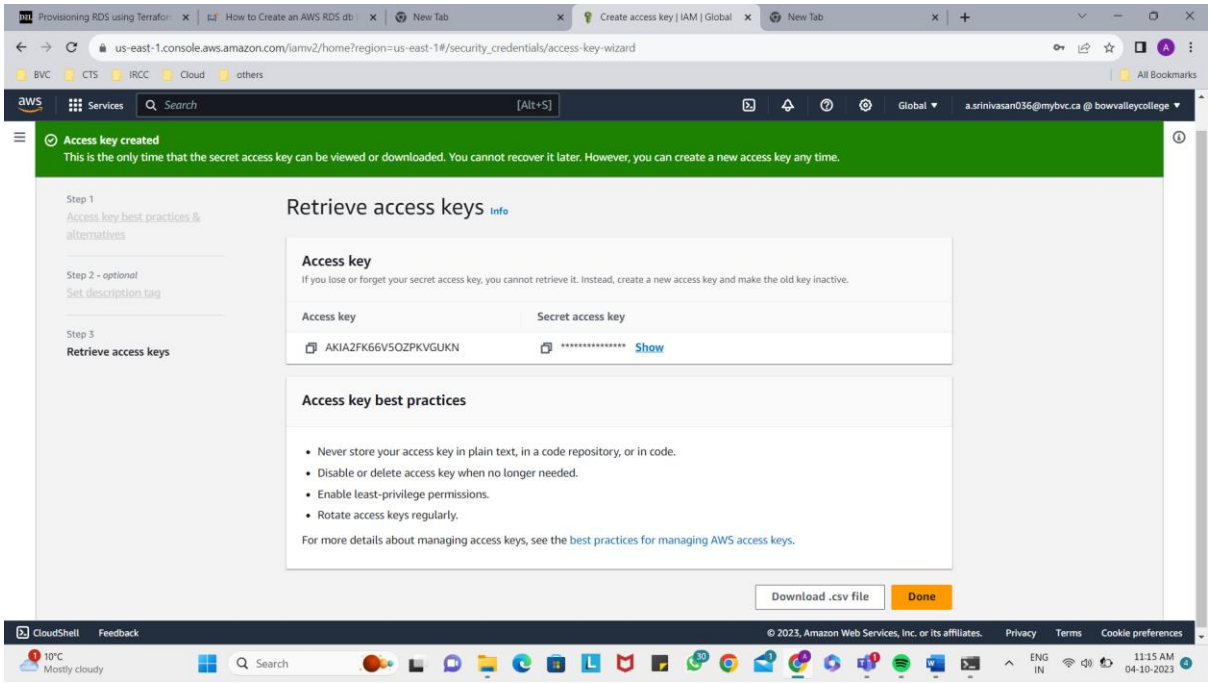
Create AWS RDS using Terraform





Create AWS RDS using Terraform

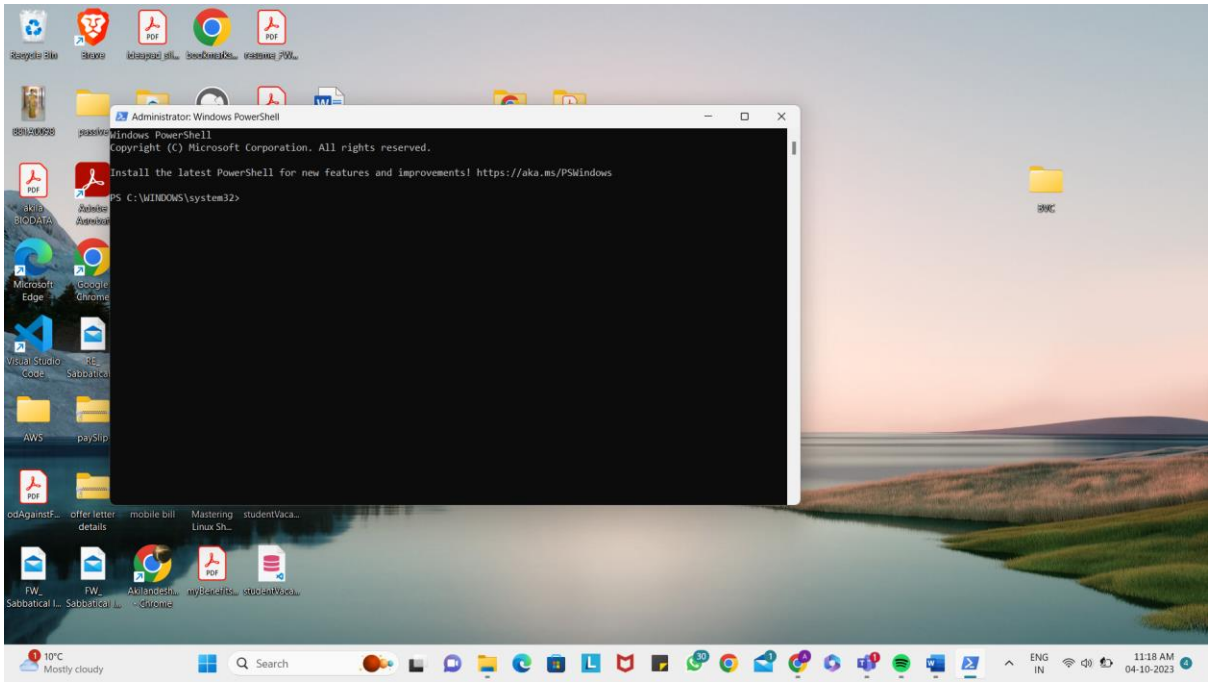


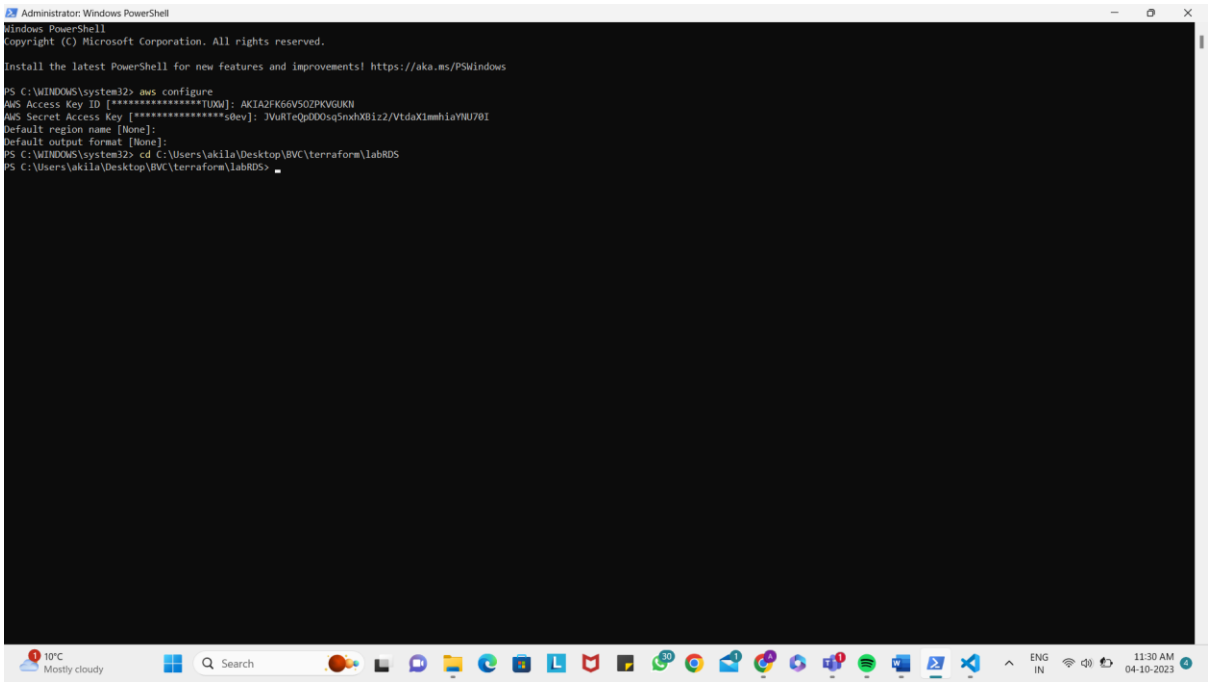


AKIA2FK66V5OTN7AHGG7

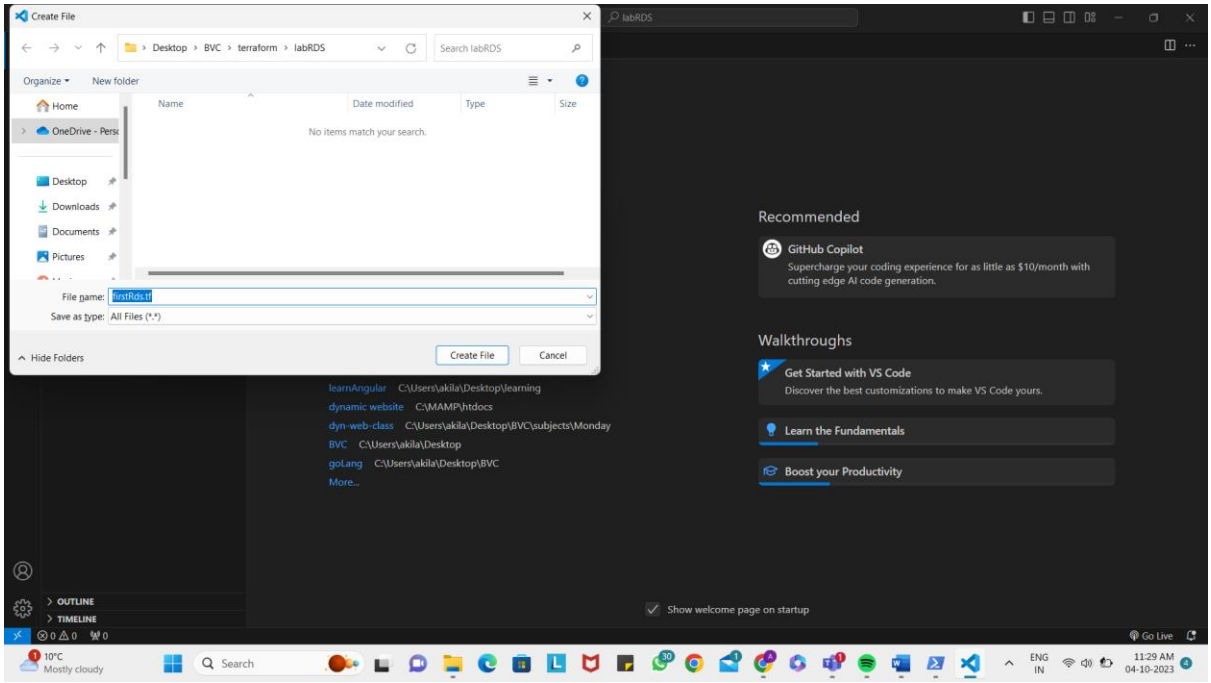
PS/ZyM0yywcnQWfNAAhFHTWfBgBdd9akcvZDzZbo

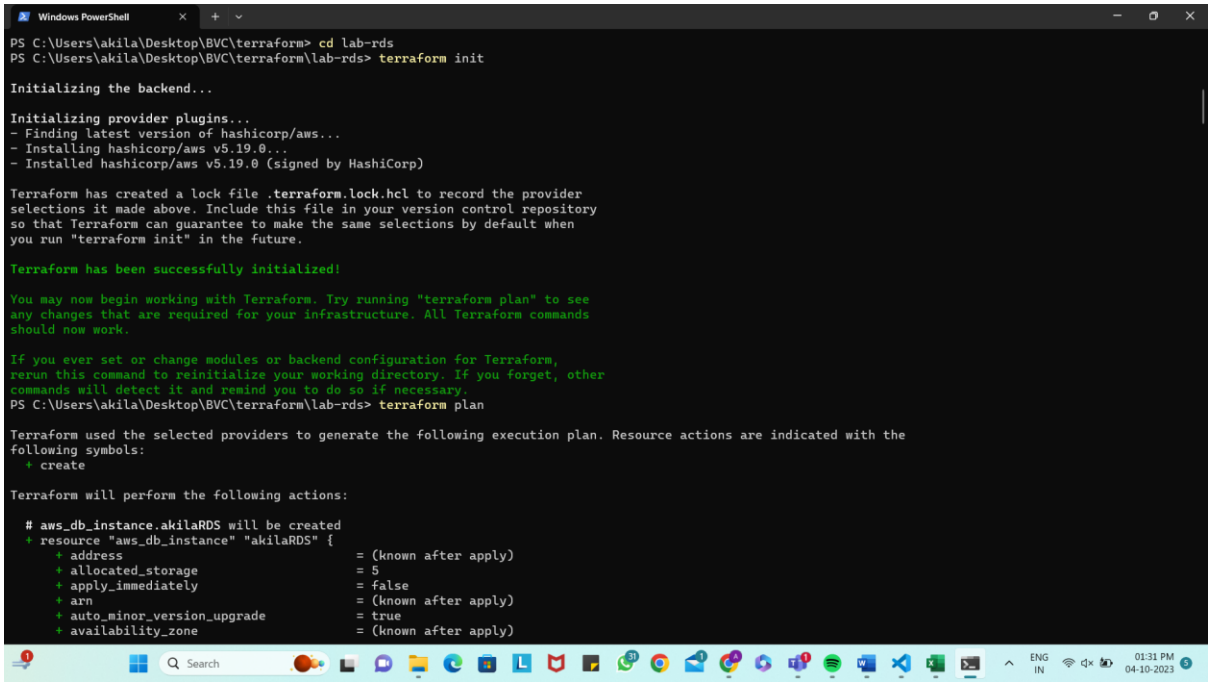
Create AWS RDS using Terraform



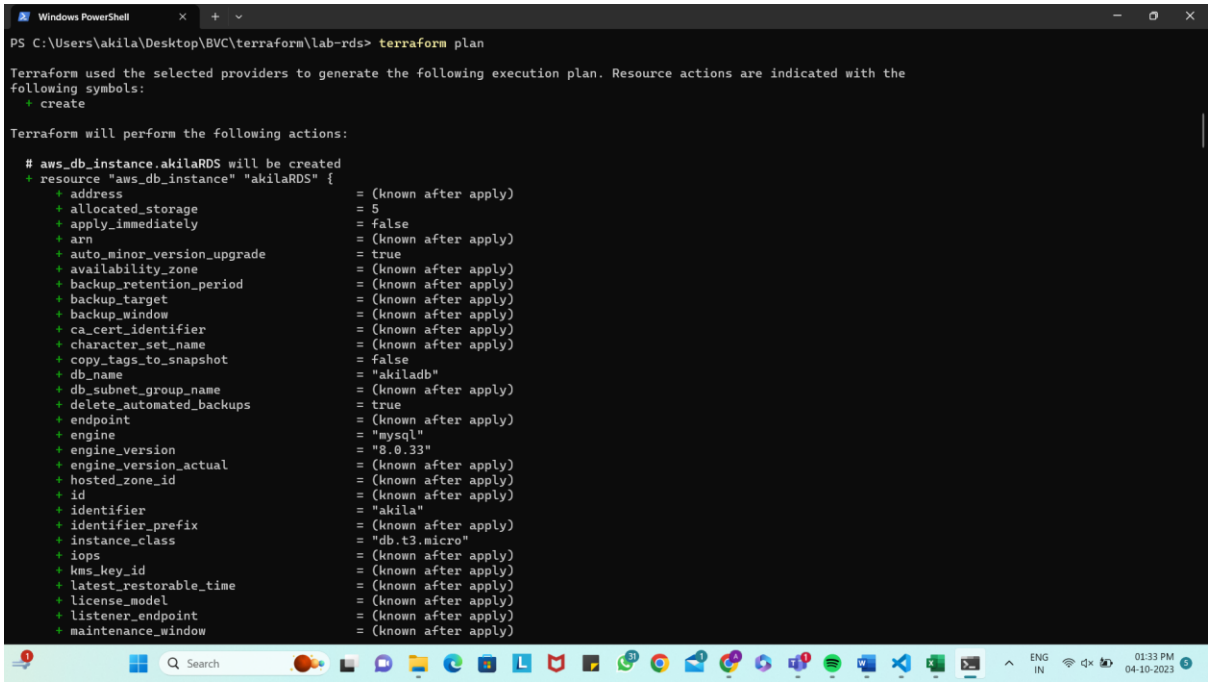


Create AWS RDS using Terraform

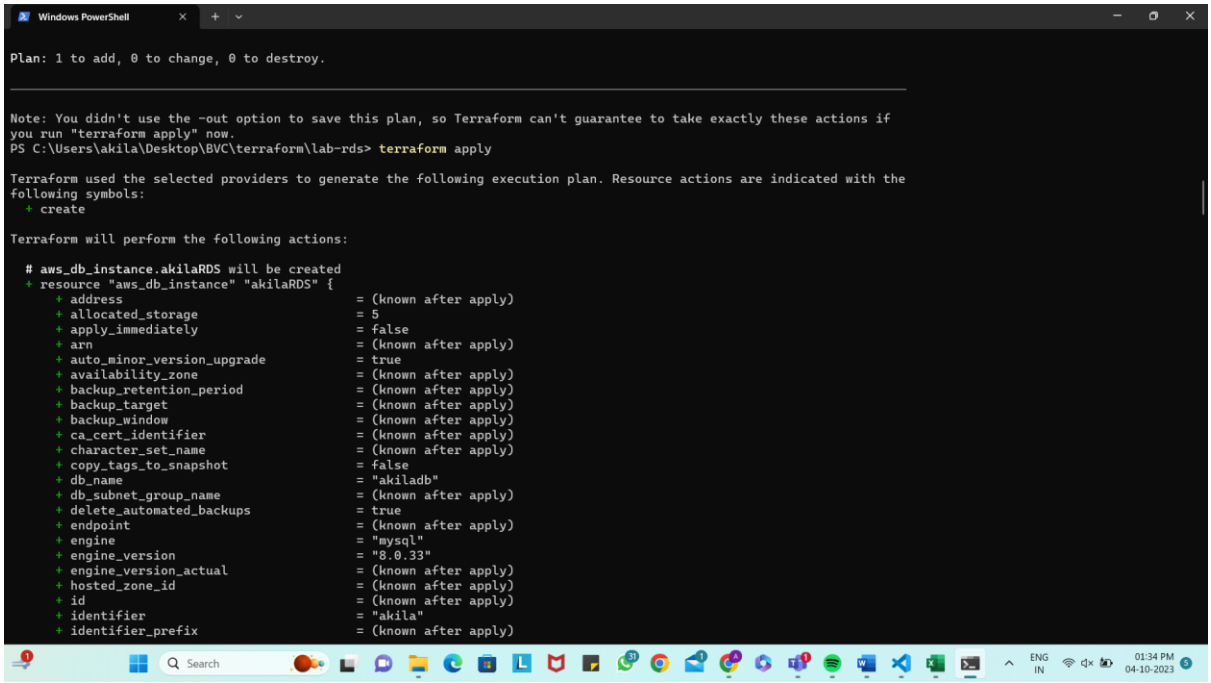




Create AWS RDS using Terraform

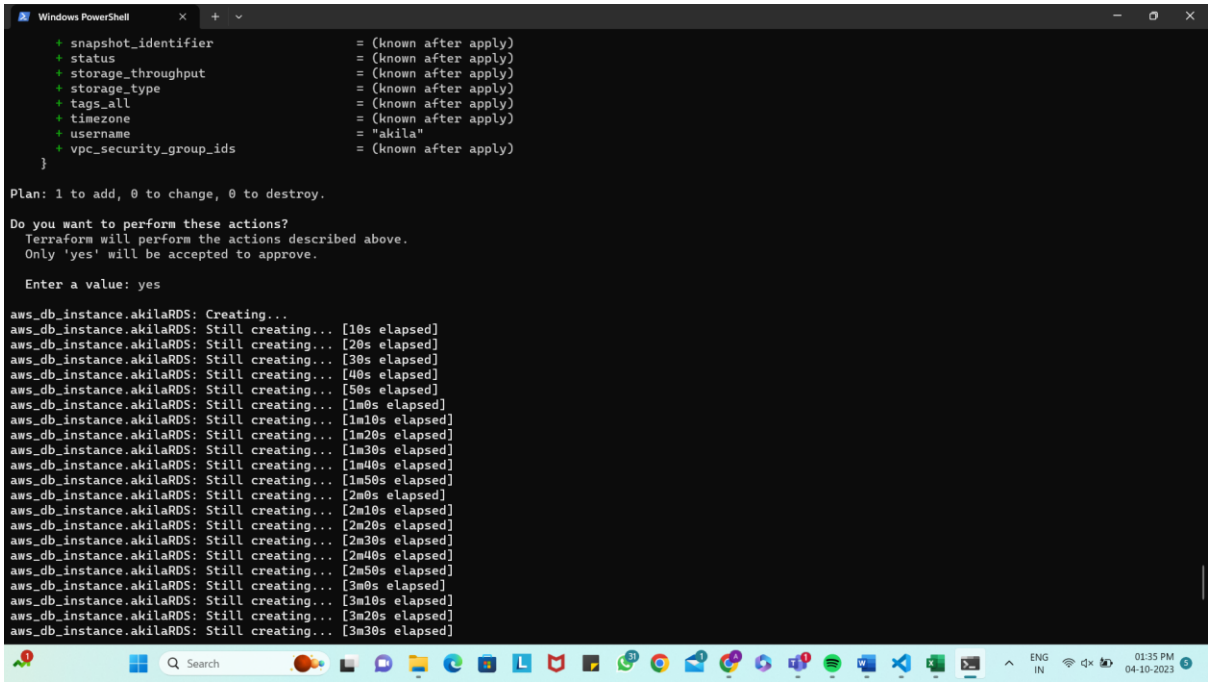


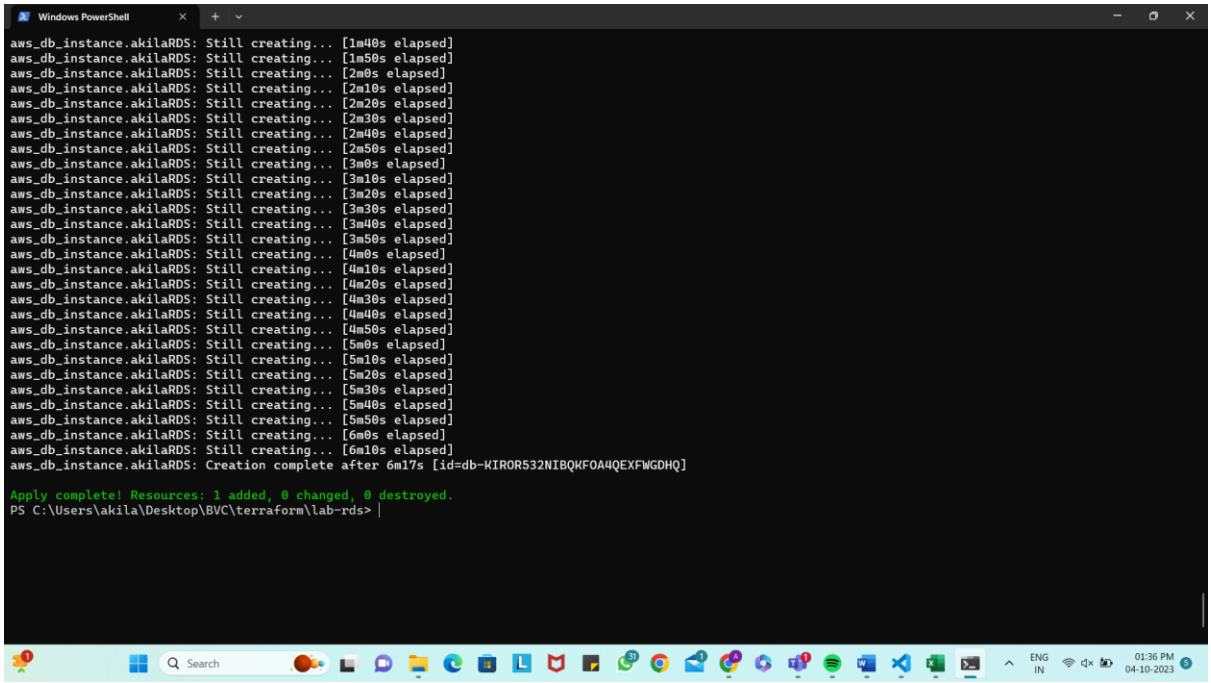
```
Windows PowerShell
PS C:\Users\akila\Desktop\BVC\terraform\lab-rds> terraform plan

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_db_instance.akilaRDS will be created
  + resource "aws_db_instance" "akilaRDS" {
      + address                               = (known after apply)
      + allocated_storage                     = 5
      + apply_immediately                     = false
      + arn                                   = (known after apply)
      + auto_minor_version_upgrade            = true
      + availability_zone                     = (known after apply)
      + backup_retention_period               = (known after apply)
      + backup_target                         = (known after apply)
      + backup_window                         = (known after apply)
      + ca_cert_identifier                    = (known after apply)
      + character_set_name                    = (known after apply)
      + copy_tags_to_snapshot                 = false
      + db_name                               = "akiladb"
      + db_subnet_group_name                  = (known after apply)
      + delete_automated_backups              = true
      + endpoint                              = (known after apply)
      + engine                                = "mysql"
      + engine_version                        = "8.0.33"
      + engine_version_actual                 = (known after apply)
      + hosted_zone_id                        = (known after apply)
      + id                                    = (known after apply)
      + identifier                            = "akila"
      + identifier_prefix                     = (known after apply)
      + instance_class                        = "db.t3.micro"
      + iops                                  = (known after apply)
      + kms_key_id                            = (known after apply)
      + latest_restorable_time                = (known after apply)
      + license_model                         = (known after apply)
      + listener_endpoint                     = (known after apply)
      + maintenance_window                    = (known after apply)
```



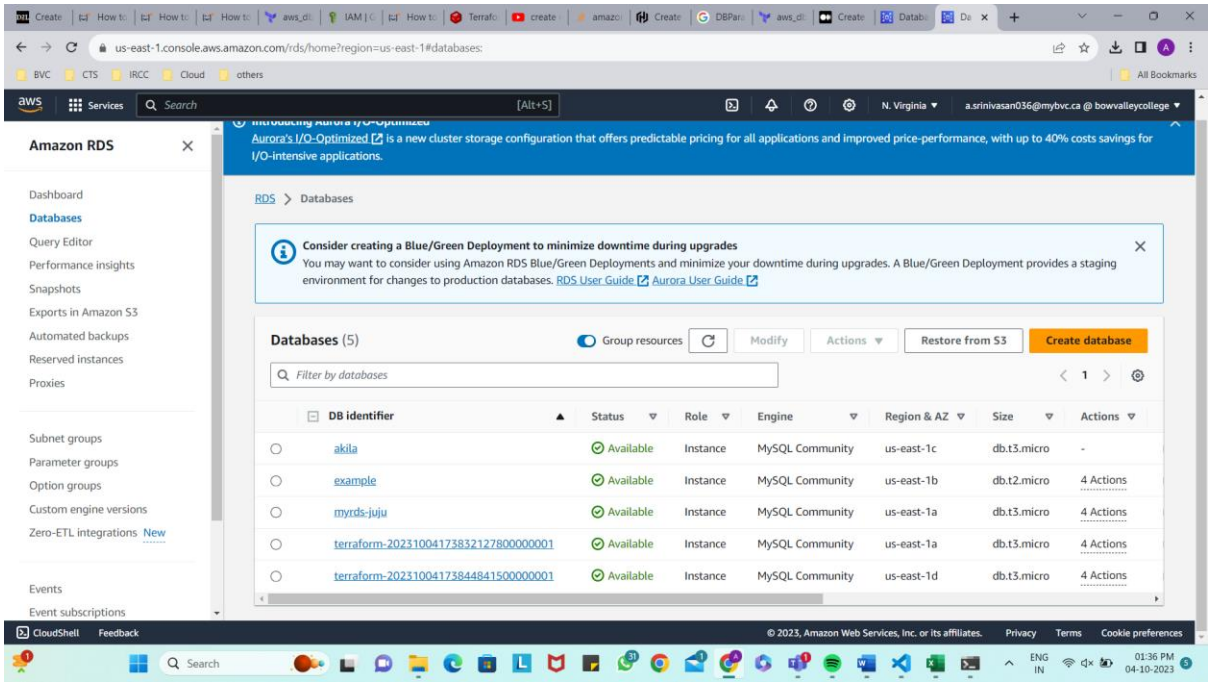
```
Windows PowerShell

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.
PS C:\Users\akila\Desktop\BVC\terraform\lab-rds> 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_db_instance.akilaRDS will be created
  + resource "aws_db_instance" "akilaRDS" {
      + address                               = (known after apply)
      + allocated_storage                     = 5
      + apply_immediately                     = false
      + arn                                   = (known after apply)
      + auto_minor_version_upgrade            = true
      + availability_zone                     = (known after apply)
      + backup_retention_period               = (known after apply)
      + backup_target                         = (known after apply)
      + backup_window                         = (known after apply)
      + ca_cert_identifier                    = (known after apply)
      + character_set_name                    = (known after apply)
      + copy_tags_to_snapshot                 = false
      + db_name                               = "akiladb"
      + db_subnet_group_name                  = (known after apply)
      + delete_automated_backups              = true
      + endpoint                              = (known after apply)
      + engine                                = "mysql"
      + engine_version                        = "8.0.33"
      + engine_version_actual                 = (known after apply)
      + hosted_zone_id                        = (known after apply)
      + id                                    = (known after apply)
      + identifier                            = "akila"
      + identifier_prefix                     = (known after apply)
```

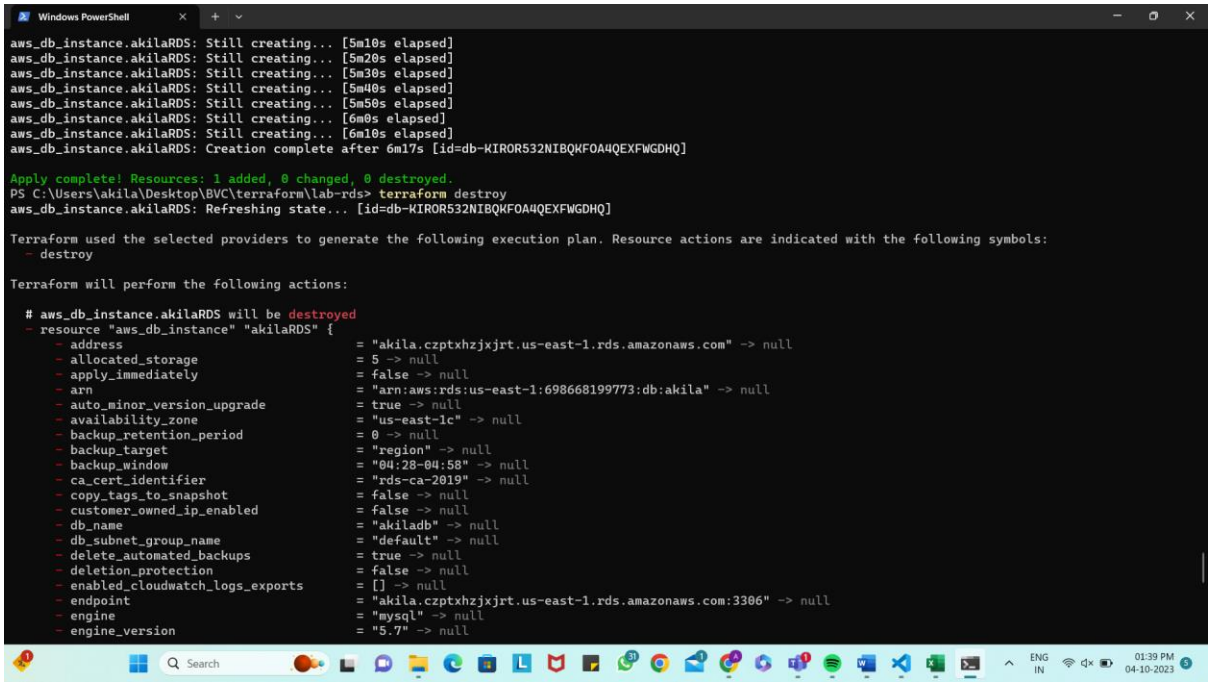
Create AWS RDS using Terraform



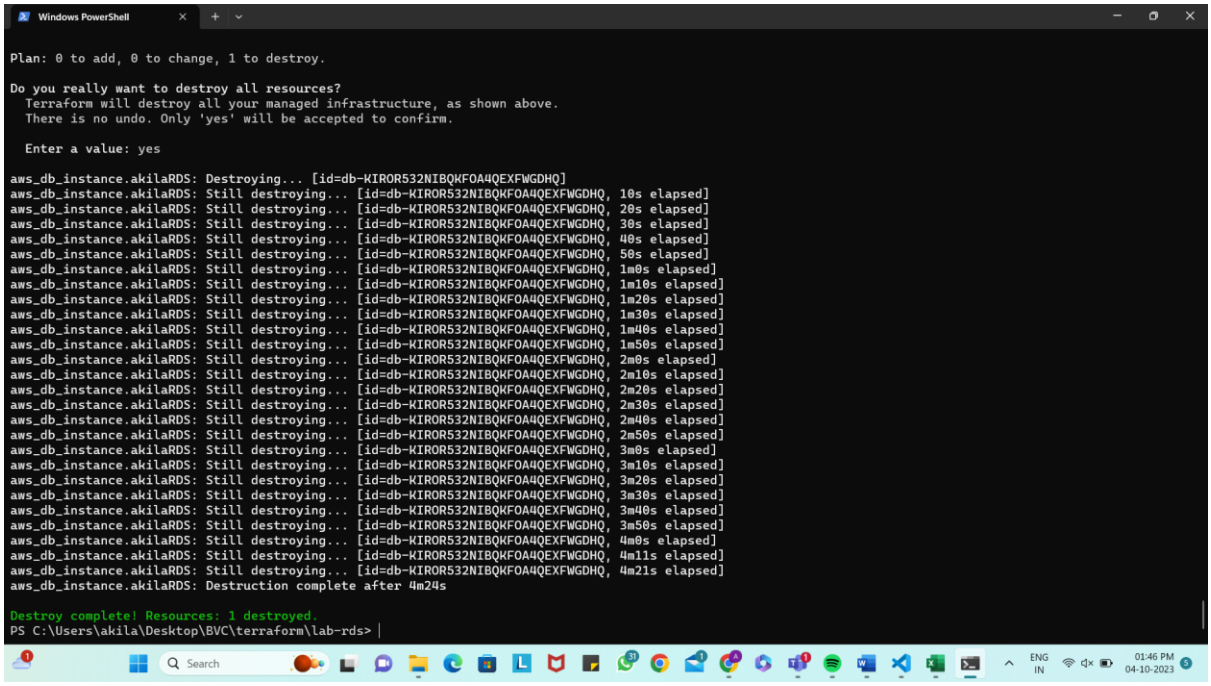


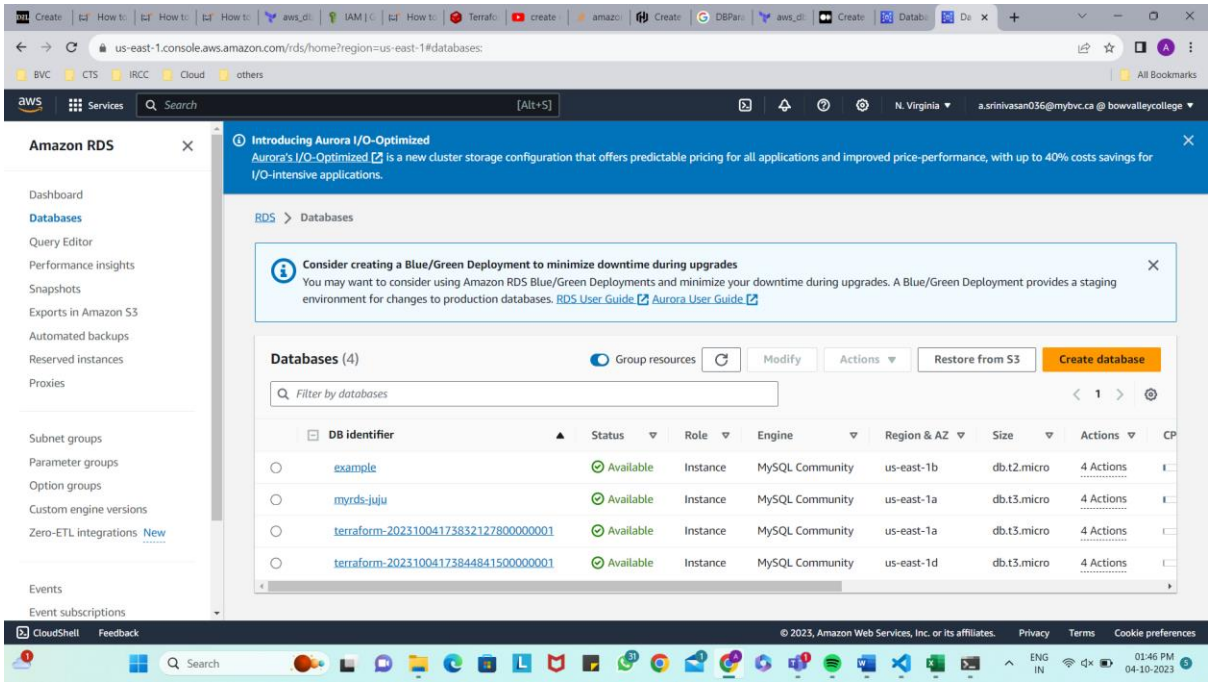
Create AWS RDS using Terraform



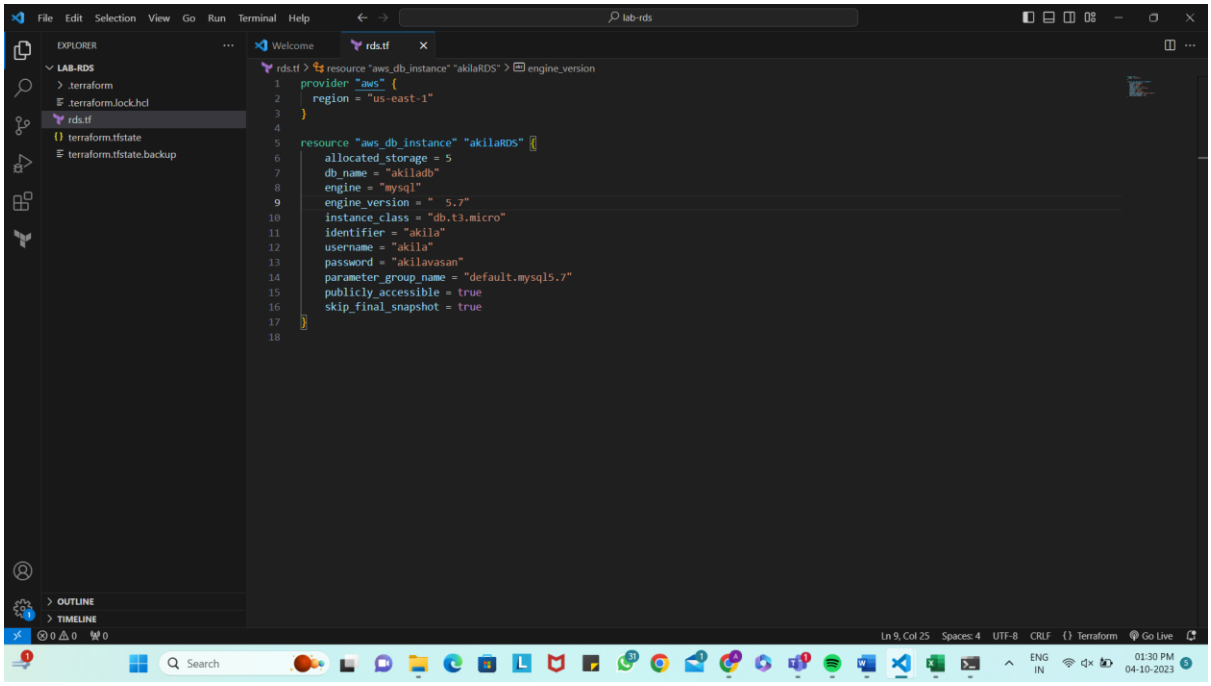


Create AWS RDS using Terraform





Create AWS RDS using Terraform



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

resource "aws_db_instance" "akilaRDS" {
    allocated_storage = 5
    db_name = "akiladb"
    engine = "mysql"
    engine_version = "  5.7"
    instance_class = "db.t3.micro"
    identifier = "akila"
    username = "akila"
    password = "akilavasan"
    parameter_group_name = "default.mysql5.7"
    publicly_accessible = true
    skip_final_snapshot = true
}
```