Lab Exercise 10– Creating an AWS RDS Instance in Terraform

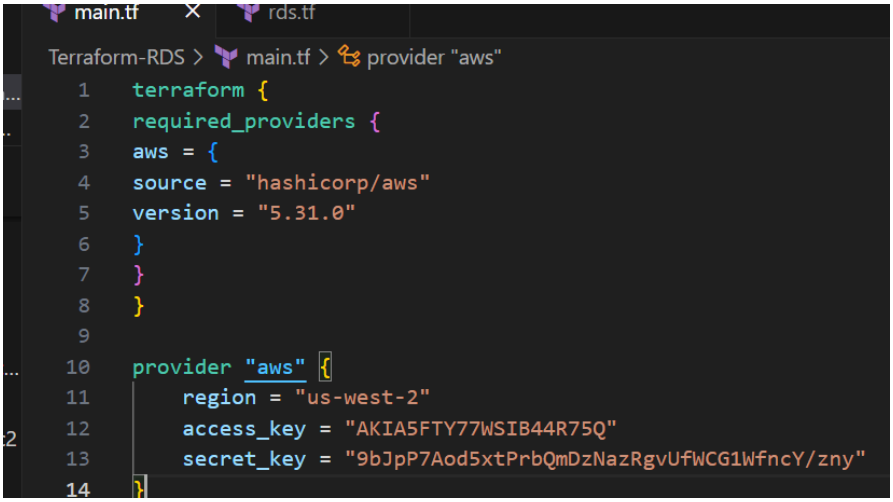
1. Create a Terraform Directory:

```
mkdir Terraform-RDS
cd Terraform-RDS
```

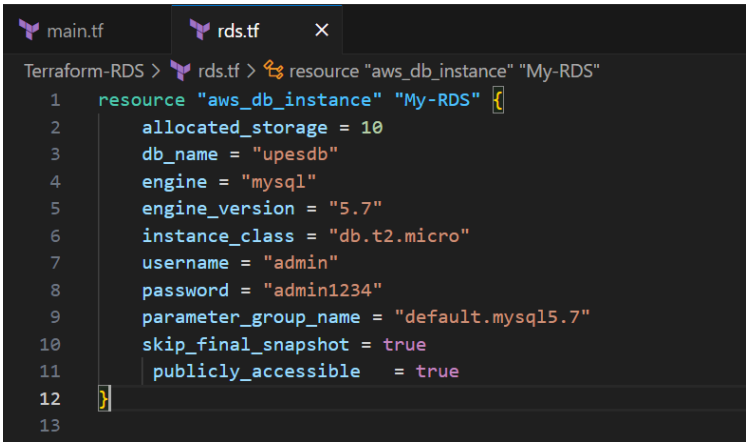
2. Create Terraform Configuration Files:

Create a file named main.tf:

 # main.tf



#rds.tf



3. Initialize and Apply:

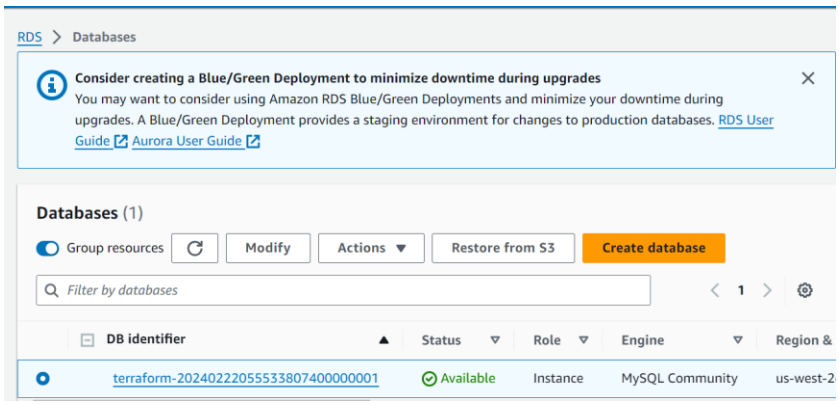
- Run the following Terraform commands to initialize and apply the configuration:

```
terraform init
terraform apply
```

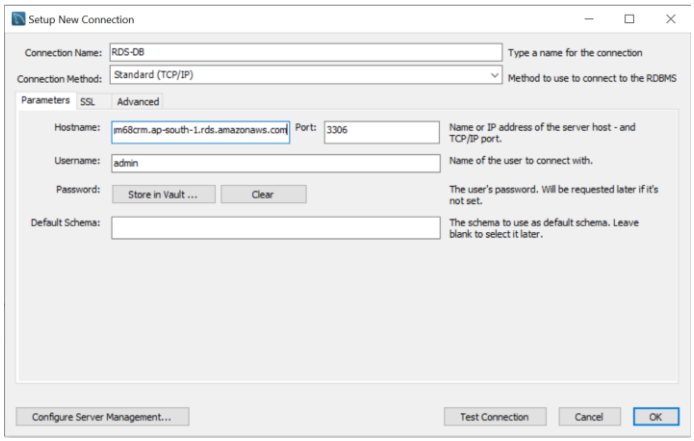
```
PS C:\Desktop\DevOps\Sem6\SMCP\Lab Files\TERRAFORM LAB SCRIPTS\Terraform-RDS> terraform init

Initializing the backend...

Initializing provider plugins...
- Finding hashicorp/aws versions matching "5.31.0"...
- Installing hashicorp/aws v5.31.0...
- Installed hashicorp/aws v5.31.0 (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
you run "terraform init" in the future.


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.
PS C:\Desktop\DevOps\Sem6\SMCP\Lab Files\TERRAFORM LAB SCRIPTS\Terraform-RDS> terraform validate
Success! The configuration is valid.
```

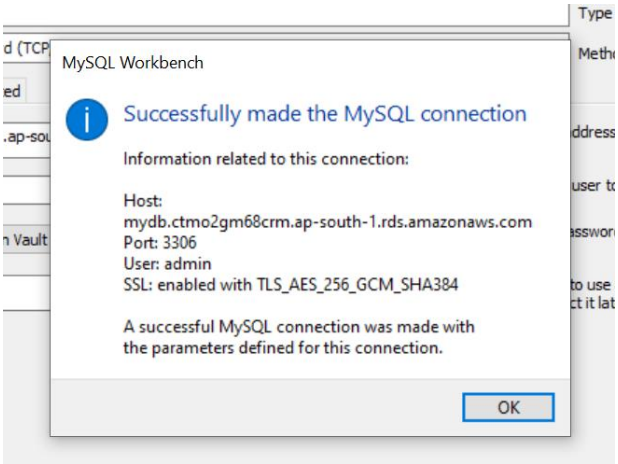
```
PS C:\Desktop\DevOps\Sem6\SMCP\Lab Files\TERRAFORM LAB SCRIPTS\Terraform-RDS> terraform apply
aws_db_instance.My-RDS: Refreshing state... [id=db-AOJ3U3CA73SPOT4EOZBC5VIIKM]

Terraform used the selected providers to generate the following execution plan. Resource actions
  + create

Terraform will perform the following actions:

  # aws_db_instance.My-RDS will be created
  + resource "aws_db_instance" "My-RDS" {
      + address                               = (known after apply)
      + allocated_storage                     = 10
      + apply_immediately                     = false
      + arn                                   = (known after apply)
      + auto_minor_version_upgrade            = true
      + availability_zone                     = (known after apply)
      + backup_retention_period               = (known after apply)
```

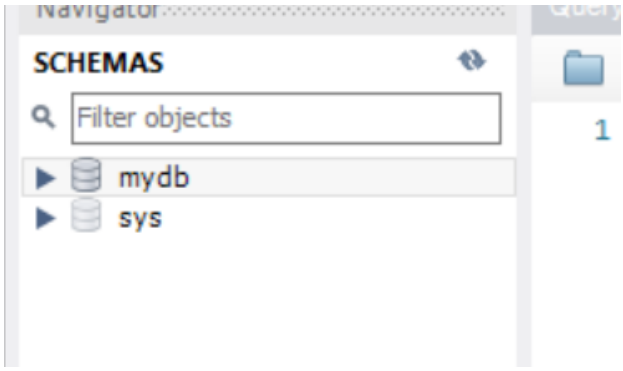
4. Verify RDS Instance in AWS Console:



5. Connect to MySQL Workbench







6. Clean Up:

terraform destroy

Confirm the destruction by typing yes.