

EXPERIMENT – 10

Aim: Creating an AWS RDS Instance in Terraform.

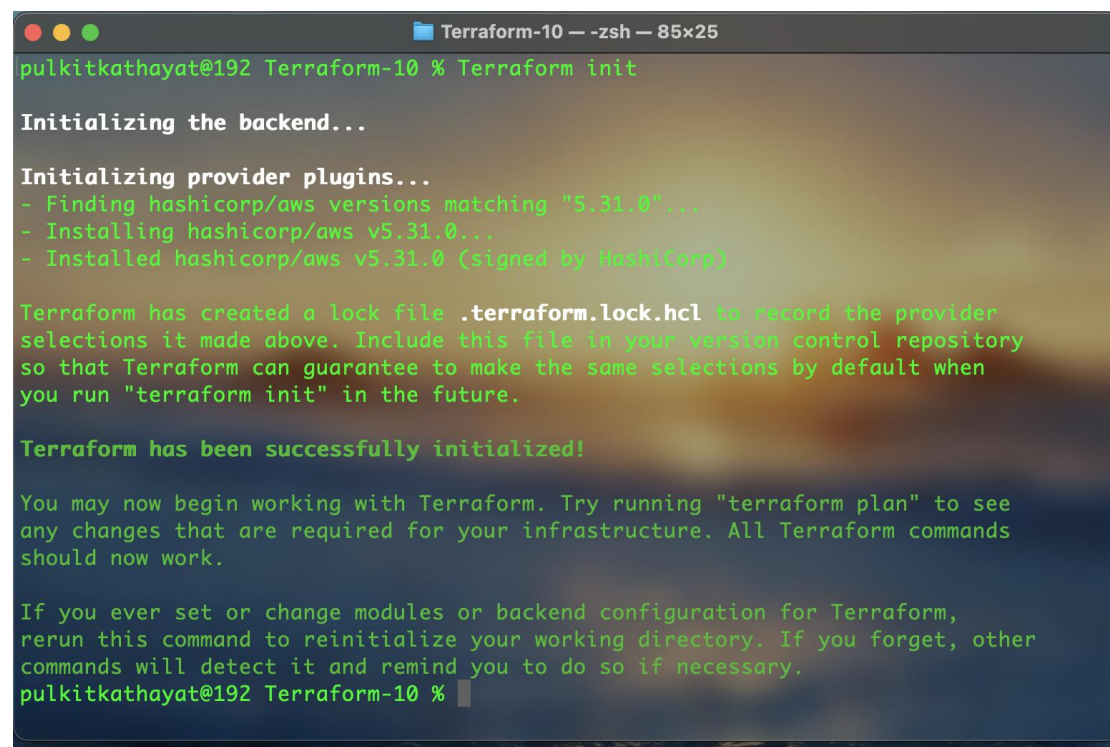
1] Create a Terraform Configuration File (main.tf)

```
main.tf × instance.tf
main.tf > terraform
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "5.31.0"
6     }
7   }
8 }
9
10 provider "aws" {
11
12   region = "ap-south-1"
13   access_key = "AKIATJHVFEM7OWRV3DM7"
14   secret_key = "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"
15 }
```

2] Create a Terraform RDS File (rds.tf)

```
Welcome rds.tf ×
rds.tf > resource "aws_db_instance" "My-RDS" > tags > Name
1 resource "aws_db_instance" "My-RDS" {
2   allocated_storage = 10
3   db_name = "upesdb"
4   engine = "mysql"
5   engine_version = "5.7"
6   instance_class = "db.t2.micro"
7   username = "admin"
8   password = "admin123"
9   parameter_group_name = "default.mysql5.7"
10  skip_final_snapshot = true
11  publicly_accessible = true
12  tags = {}
13    Name = "My-RDS"
14  }
15
16 }
```

3] Initialize Terraform using command “terraform init”

A terminal window titled "Terraform-10 — -zsh — 85x25" showing the execution of the 'terraform init' command. The output indicates that the backend is being initialized, followed by the installation of the AWS provider plugin. It mentions finding and installing hashicorp/aws v5.31.0. A lock file '.terraform.lock.hcl' is created to record these selections. The process concludes with a success message and instructions on how to proceed with 'terraform plan' and reinitialization if needed.

```
pulkitkathayat@192 Terraform-10 % 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)

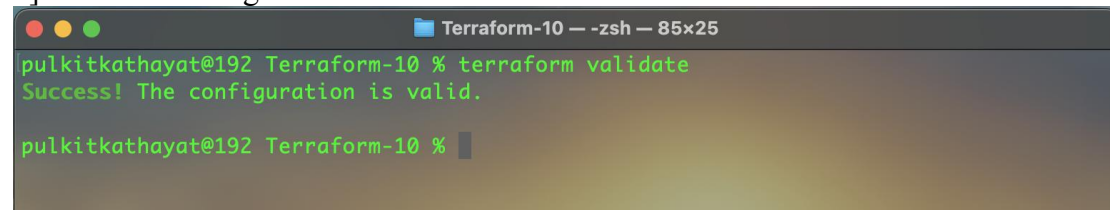
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

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.
pulkitkathayat@192 Terraform-10 %
```

4] Validate it using command “terraform validate”

A terminal window titled "Terraform-10 — -zsh — 85x25" showing the execution of the 'terraform validate' command. The output is a simple success message stating that the configuration is valid.

```
pulkitkathayat@192 Terraform-10 % terraform validate
Success! The configuration is valid.

pulkitkathayat@192 Terraform-10 %
```

5] Check the Plan using command “terraform plan”

```
pulkitkathayat@192 Terraform-10 % 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.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)
  + 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                               = "upesdb"
  + db_subnet_group_name                  = (known after apply)
  + delete_automated_backups              = true
  + endpoint                             = (known after apply)
  + engine                                = "mysql"
  + engine_version                        = "5.7"
  + engine_version_actual                  = (known after apply)
  + hosted_zone_id                        = (known after apply)
  + id                                    = (known after apply)
  + identifier                            = (known after apply)
  + identifier_prefix                     = (known after apply)
  + instance_class                        = "db.t2.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)
  + master_user_secret                    = (known after apply)
  + master_user_secret_kms_key_id         = (known after apply)
  + monitoring_interval                    = 0
  + monitoring_role_arn                   = (known after apply)
  + multi_az                              = (known after apply)
```

```

+ multi_az                = (known after apply)
+ nchar_character_set_name = (known after apply)
+ network_type            = (known after apply)
+ option_group_name       = (known after apply)
+ parameter_group_name    = "default.mysql5.7"
+ password                = (sensitive value)
+ performance_insights_enabled = false
+ performance_insights_kms_key_id = (known after apply)
+ performance_insights_retention_period = (known after apply)
+ port                    = (known after apply)
+ publicly_accessible     = true
+ replica_mode            = (known after apply)
+ replicas                = (known after apply)
+ resource_id             = (known after apply)
+ skip_final_snapshot     = true
+ snapshot_identifier     = (known after apply)
+ status                  = (known after apply)
+ storage_throughput      = (known after apply)
+ storage_type            = (known after apply)
+ tags                    = {
  + "Name" = "My-RDS"
}
+ tags_all                = {
  + "Name" = "My-RDS"
}
+ timezone                = (known after apply)
+ username                = "admin"
+ vpc_security_group_ids  = (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.

pulkitkathayat@192 Terraform-10 %

6] Apply it using command “Terraform apply -auto-approve”

```
pulkitkathayat@192 Terraform-10 % Terraform apply -auto-approve
```

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.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)
  + 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                               = "upesdb"
  + db_subnet_group_name                  = (known after apply)
  + delete_automated_backups              = true
  + endpoint                             = (known after apply)
  + engine                                = "mysql"
  + engine_version                        = "5.7"
  + engine_version_actual                  = (known after apply)
  + hosted_zone_id                        = (known after apply)
  + id                                    = (known after apply)
  + identifier                            = (known after apply)
  + identifier_prefix                      = (known after apply)
  + instance_class                        = "db.t2.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)
  + master_user_secret                     = (known after apply)
  + master_user_secret_kms_key_id         = (known after apply)
  + monitoring_interval                    = 0
  + monitoring_role_arn                    = (known after apply)
  + multi_az                              = (known after apply)
  + nchar_character_set_name              = (known after apply)
  + network_type                          = (known after apply)
  + option_group_name                     = (known after apply)
```



```

}

Plan: 1 to add, 0 to change, 0 to destroy.
aws_db_instance.My-RDS: Creating...
aws_db_instance.My-RDS: Still creating... [10s elapsed]
aws_db_instance.My-RDS: Still creating... [20s elapsed]
aws_db_instance.My-RDS: Still creating... [30s elapsed]
aws_db_instance.My-RDS: Still creating... [40s elapsed]
aws_db_instance.My-RDS: Still creating... [50s elapsed]
aws_db_instance.My-RDS: Still creating... [1m0s elapsed]
aws_db_instance.My-RDS: Still creating... [1m10s elapsed]
aws_db_instance.My-RDS: Still creating... [1m20s elapsed]
aws_db_instance.My-RDS: Still creating... [1m30s elapsed]
aws_db_instance.My-RDS: Still creating... [1m40s elapsed]
aws_db_instance.My-RDS: Still creating... [1m50s elapsed]
aws_db_instance.My-RDS: Still creating... [2m0s elapsed]
aws_db_instance.My-RDS: Still creating... [2m10s elapsed]
aws_db_instance.My-RDS: Still creating... [2m20s elapsed]
aws_db_instance.My-RDS: Still creating... [2m30s elapsed]
aws_db_instance.My-RDS: Still creating... [2m40s elapsed]
aws_db_instance.My-RDS: Still creating... [2m50s elapsed]
aws_db_instance.My-RDS: Still creating... [3m0s elapsed]
aws_db_instance.My-RDS: Still creating... [3m10s elapsed]
aws_db_instance.My-RDS: Still creating... [3m20s elapsed]
aws_db_instance.My-RDS: Still creating... [3m30s elapsed]
aws_db_instance.My-RDS: Still creating... [3m40s elapsed]
aws_db_instance.My-RDS: Still creating... [3m50s elapsed]
aws_db_instance.My-RDS: Still creating... [4m0s elapsed]
aws_db_instance.My-RDS: Still creating... [4m10s elapsed]
aws_db_instance.My-RDS: Still creating... [4m20s elapsed]
aws_db_instance.My-RDS: Creation complete after 4m28s [id=db-2UZYCGYQOWXTJNVF52APRUTA4E]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
pulkitkathayat@192 Terraform-10 %

```

7] Verify Resources on AWS Management Console.

The screenshot shows the AWS Management Console interface for Amazon RDS. On the left is a navigation menu with options like Dashboard, Databases, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, and Custom engine versions. The main content area is titled 'Databases (1)' and shows a table with the following data:

DB identifier	Status	Role	Engine	Region & AZ	Size	Rec
terraform-20240225162338721900000001	Available	Instance	MySQL Community	ap-south-1b	db.t2.micro	

At the top of the console, there is a notification about 'Introducing Aurora I/O-Optimized' and a tip about 'Consider creating a Blue/Green Deployment to minimize downtime during upgrades'.

8] Connect with MySQL Workbench with proper Configuration and save it.

Enter a value: yes

```
aws_db_instance.My-RDS: Destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 10s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 20s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 30s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 40s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 50s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 1m0s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 1m10s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 1m20s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 1m30s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 1m40s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 1m50s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 2m0s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 2m10s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 2m20s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 2m30s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 2m40s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 2m50s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 3m0s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 3m10s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 3m20s elapsed]

aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 3m30s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 3m40s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 3m50s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 4m0s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-ZUZYCGYQOWXTJNVFS2APRUTA4E, 4m10s elapsed]
```