

## Lab Exercise 10– Creating an AWS RDS Instance in Terraform

### Objective:

Learn how to use Terraform to create an AWS RDS instance.

### Prerequisites:

- Terraform installed on your machine.
- AWS CLI configured with the necessary credentials.

### Steps:

#### 1. Create a Terraform Directory:

```
mkdir terraform-rds
cd terraform-rds
```

#### 2. Create Terraform Configuration Files:

Create a file named main.tf:

# main.tf

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

resource "aws_db_instance" "My-RDS" {
  allocated_storage = 10
  db_name = "upesdb"
  engine = "mysql"
```

```
engine_version = "5.7"
instance_class = "db.t2.micro"
username = "admin"
password = "Hitesh111"
parameter_group_name = "default.mysql5.7"
skip_final_snapshot = true
}
```

- Replace "YourPassword123" with a secure password and "your-security-group-id" with your actual security group ID.
- In this configuration, we define an AWS RDS instance with specific settings, such as engine type, instance class, and security group.

### 3. Initialize and Apply:

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

```
terraform init
terraform apply
```

- Terraform will prompt you to confirm the creation of the RDS instance. Type yes and press Enter.

### 4. Verify RDS Instance in AWS Console:

- Log in to the AWS Management Console and navigate to the RDS service.
- Verify that the specified RDS instance with the specified settings has been created.

### 5. Update RDS Configuration:

- If you want to modify the RDS instance configuration, update the main.tf file with the desired changes.
- Rerun the terraform apply command to apply the changes:

```
terraform apply
```

## 6. Clean Up:

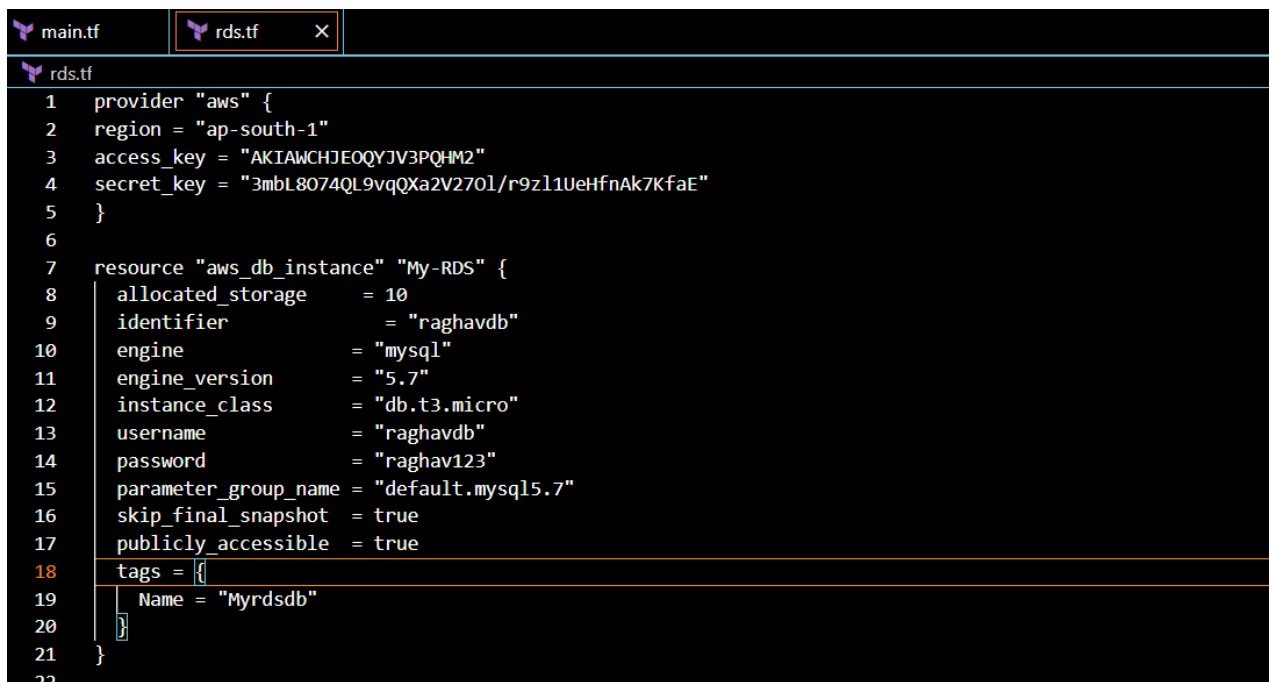
After testing, you can clean up the RDS instance:

```
terraform destroy
```

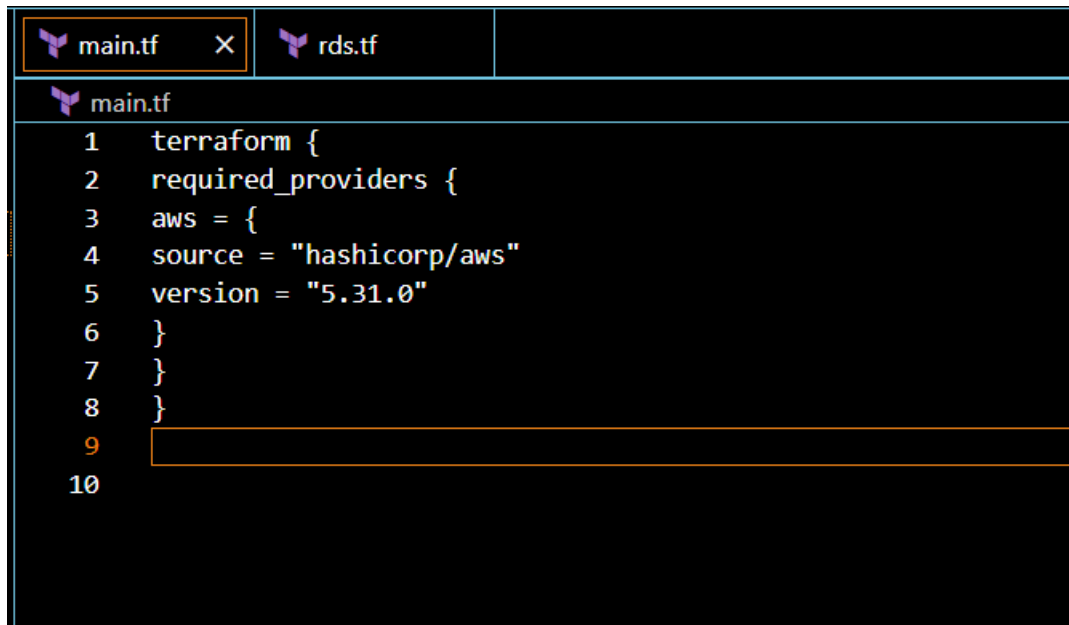
Confirm the destruction by typing yes.

## 7. Conclusion:

This lab exercise demonstrates how to use Terraform to create an AWS RDS instance. You learned how to define RDS settings, initialize and apply the Terraform configuration, and verify the creation of the RDS instance in the AWS Management Console. Experiment with different RDS settings in the main.tf file to observe how

A screenshot of a code editor with a dark theme. At the top, there are two tabs: 'main.tf' and 'rds.tf'. The 'rds.tf' tab is active and highlighted with a yellow border. Below the tabs, the content of 'rds.tf' is displayed. The code defines an AWS provider and an AWS RDS instance resource. Line numbers 1 through 22 are visible on the left side of the editor. The code is as follows:

```
1 provider "aws" {
2   region = "ap-south-1"
3   access_key = "AKIAWCHJEOQYJV3PQHM2"
4   secret_key = "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"
5 }
6
7 resource "aws_db_instance" "My-RDS" {
8   allocated_storage = 10
9   identifier        = "raghavdb"
10  engine             = "mysql"
11  engine_version     = "5.7"
12  instance_class     = "db.t3.micro"
13  username           = "raghavdb"
14  password           = "raghav123"
15  parameter_group_name = "default.mysql5.7"
16  skip_final_snapshot = true
17  publicly_accessible = true
18  tags = [{
19    Name = "Myrdsdb"
20  }]
21 }
22
```



```
main.tf x rds.tf
main.tf
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "5.31.0"
6     }
7   }
8 }
9
10
```

```
PS C:\Users\anu39\Terraform-Script> 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.

```
PS C:\Users\anu39\Terraform-Script> terraform validate
Success! The configuration is valid.
```

```
PS C:\Users\anu39\Terraform-Script> 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                      = (known after apply)
  + 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                   = "raghavdb"
  + 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)
```

```
  + 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)
  + 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" = "Myrdsdb"
  }
  + tags_all                     = {
    + "Name" = "Myrdsdb"
  }
  + timezone                    = (known after apply)
  + username                    = "raghavdb"
  + 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.

```
PS C:\Users\anu39\Terraform-Script> 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                               = (known after apply)
  + 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                           = "raghavdb"
  + 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)
}
```

```
+ 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)
+ 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" = "Myrdsdb"
}
+ tags_all                                = {
  + "Name" = "Myrdsdb"
}
+ timezone                                = (known after apply)
+ username                                = "raghavdb"
+ vpc_security_group_ids                  = (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: Creation complete after 3m55s [id=db-5MFTCL7JYNS3QWDBJB44EPRRY]
```

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

PS C:\Users\anu39\Terraform-Script>

RDS > Databases

Consider creating a Blue/Green Deployment to minimize downtime during upgrades  
You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. [RDS User Guide](#) [Aurora User Guide](#)

Databases (1) ☒ Group resources

Filter by databases

<input type="checkbox"/> DB identifier ▲	Status ▼	Role ▼	Engine ▼	Region & AZ ▼	Size ▼	Recommendations ▼	CPU
raghavdb	Available	Instance	MySQL Community	ap-south-1c	db.t3.micro		-

DB identifier: raghavdb  
CPU: 19.24%

Status: Available  
Class: db.t3.micro

Role: Instance  
Current activity: 0 Connections

Engine: MySQL Community  
Region & AZ: ap-south-1c

Recommendations

Connectivity & security | Monitoring | Logs & events | Configuration | Maintenance & backups | Tags | Recommendations

### Connectivity & security

Endpoint & port	Networking	Security
<b>Endpoint</b> raghavdb.cxgqys8yovp8.ap-south-1.rds.amazonaws.com	<b>Availability Zone</b> ap-south-1c	<b>VPC security groups</b> default (sg-03cc52f2e5e1b06eb) Active
<b>Port</b> 3306	<b>VPC</b> vpc-0e4fe532fd5496e97	<b>Publicly accessible</b> Yes
	<b>Subnet group</b> default	<b>Certificate authority</b> Info rds-ca-rsa2048-g1
	<b>Subnets</b>	

Setup New Connection

Connection Name:  Type a name for the connection

Connection Method:  Method to use to connect to the RDBMS

Parameters SSL Advanced

Hostname:  Port:  Name or IP address of the server host - and TCP/IP port.

Username:  Name of the user to connect with.

Password:   The user's password. Will be requested later if it's not set.

Default Schema:  The schema to use as default schema. Leave blank to select it later.

Store Password For Connection

Please enter password for the following service:

Service: Mysql@raghavdb.cxgqys8yovp8.ap-south-1.rds.amazonaws.com:3306

User: raghavdb

Password:

MySQL Workbench


**Successfully made the MySQL connection**


Information related to this connection:

Host:  
raghavdb.cxgqys8yovp8.ap-south-1.rds.amazonaws.com  
Port: 3306  
User: raghavdb  
SSL: enabled with ECDHE-RSA-AES128-GCM-SHA256

A successful MySQL connection was made with the parameters defined for this connection.

MyRDS

 raghavdb

 raghavdb.cxgqys8yovp8.ap-south-1....



#	Time	Action	Message	Duration / Fetch
1	22:22:35	create database rds	1 row(s) affected	0.031 sec
2	22:22:47	use rds	0 row(s) affected	0.031 sec
3	22:24:22	create table students( stu_id int stu_name varchar(25) )	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server versi...	0.047 sec
4	22:25:08	CREATE TABLE students ( stu_id INT, stu_name VARCHAR(25) stu_city VARCHAR(25) )	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server versi...	0.032 sec
5	22:25:43	CREATE TABLE students ( stu_id INT, stu_name VARCHAR(25) stu_city VARCHAR(25) )	0 row(s) affected	0.047 sec
6	22:25:54	show tables	1 row(s) returned	0.031 sec / 0.000 sec
7	22:26:15	show " from students	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server versi...	0.031 sec
8	22:26:49	SELECT * FROM students	0 row(s) returned	0.047 sec / 0.000 sec
9	22:33:10	INSERT INTO students VALUES (500094899, 'Raghav Mittal', 'Saharanpur')	1 row(s) affected	0.031 sec
10	22:33:10	INSERT INTO students VALUES (500094898, 'Keshav Bhardwaj', 'Dehradun')	1 row(s) affected	0.047 sec
11	22:33:10	INSERT INTO students VALUES (500095575, 'Ayush Singh Kushwah', 'Dehradun')	1 row(s) affected	0.032 sec
12	22:33:10	INSERT INTO students VALUES (500095286, 'Vidhant Maan Thapa', 'Dehradun')	1 row(s) affected	0.031 sec
13	22:33:10	INSERT INTO students VALUES (500095605, 'Rohin Mehrotra', 'Delhi')	1 row(s) affected	0.031 sec
14	22:33:10	INSERT INTO students VALUES (500095433, 'Aniket Bhardwaj', 'Karnal')	1 row(s) affected	0.031 sec
15	22:33:10	INSERT INTO students VALUES (500095268, 'Chitranshu Shama', 'Ghaziabad')	1 row(s) affected	0.032 sec
16	22:33:10	INSERT INTO students VALUES (500095191, 'Devesh Singh', 'Pratapgarh')	1 row(s) affected	0.046 sec
17	22:33:49	Select *from students	8 row(s) returned	0.031 sec / 0.000 sec

1 • Select \*from students;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	stu_id	stu_name	stu_city
▶	500094899	Raghav Mittal	Saharanpur
	500094898	Keshav Bhardwaj	Dehradun
	500095575	Ayush Singh Kushwah	Dehradun
	500095286	Vidhant Maan Thapa	Dehradun
	500095605	Rohin Mehrotra	Delhi
	500095433	Aniket Bhardwaj	Karnal
	500095268	Chitranshu Sharma	Ghaziabad
	500095191	Devesh Singh	Pratapgarh

RDS > Databases > raghavdb				
raghavdb				
Summary				
DB identifier raghavdb	Status Available	Role Instance	Engine MySQL Community	Recommendations
CPU 2.57%	Class db.t3.micro	Current activity 2 Connections	Region & AZ ap-south-1c	
Connectivity & security   Monitoring   Logs & events   Configuration   Maintenance & backups   Tags   Recommendations				
Connectivity & security				
Endpoint & port raghavdb.cxgqys8yovp8.ap-south-	Networking Availability Zone ap-south-1c	Security VPC security groups default (sg-03cc52f2e5e1b06eb)		

aws\_db\_instance.My-RDS: Refreshing state... [id=db-5MFTCL7JYNS3QWDBJB44EPRRY]

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:

```
# aws_db_instance.My-RDS will be destroyed
- resource "aws_db_instance" "My-RDS" {
  - address                               = "raghavdb.cxgqys8yovp8.ap-south-1.rds.amazonaws.com" -> null
  - allocated_storage                     = 10 -> null
  - apply_immediately                     = false -> null
  - arn                                   = "arn:aws:rds:ap-south-1:417100756016:db:raghavdb" -> null
  - auto_minor_version_upgrade            = true -> null
  - availability_zone                     = "ap-south-1c" -> null
  - backup_retention_period                = 0 -> null
  - backup_target                         = "region" -> null
  - backup_window                         = "22:47-23:17" -> null
  - ca_cert_identifier                    = "rds-ca-rsa2048-g1" -> null
  - copy_tags_to_snapshot                 = false -> null
  - customer_owned_ip_enabled             = false -> null
  - db_subnet_group_name                  = "default" -> null
  - delete_automated_backups              = true -> null
  - deletion_protection                   = false -> null
  - enabled_cloudwatch_logs_exports        = [] -> null
  - endpoint                              = "raghavdb.cxgqys8yovp8.ap-south-1.rds.amazonaws.com:3306" -> null
  - engine                                = "mysql" -> null
  - engine_version                        = "5.7" -> null
  - engine_version_actual                  = "5.7.44" -> null
  - hosted_zone_id                        = "Z2VFMSZA74J7XZ" -> null
  - iam_database_authentication_enabled    = false -> null
  - id                                    = "db-5MFTCL7JYNS3QWDBJB44EPRRY" -> null
  - identifier                            = "raghavdb" -> null
  - instance_class                         = "db.t3.micro" -> null
  - iops                                  = 0 -> null
  - license_model                         = "general-public-license" -> null
  - listener_endpoint                     = [] -> null
  - maintenance_window                    = "sun:13:07-sun:13:37" -> null
  - master_user_secret                     = [] -> null
```

```
- master_user_secret                     = [] -> null
- max_allocated_storage                  = 0 -> null
- monitoring_interval                     = 0 -> null
- multi_az                               = false -> null
- network_type                           = "IPv4" -> null
- option_group_name                       = "default:mysql-5-7" -> null
- parameter_group_name                    = "default:mysql5.7" -> null
- password                               = (sensitive value) -> null
- performance_insights_enabled            = false -> null
- performance_insights_retention_period   = 0 -> null
- port                                    = 3306 -> null
- publicly_accessible                     = true -> null
- replicas                                = [] -> null
- resource_id                             = "db-5MFTCL7JYNS3QWDBJB44EPRRY" -> null
- skip_final_snapshot                     = true -> null
- status                                  = "available" -> null
- storage_encrypted                       = false -> null
- storage_throughput                       = 0 -> null
- storage_type                             = "gp2" -> null
- tags                                    = {
  - "Name" = "Myrdsdb"
} -> null
- tags_all                                = {
  - "Name" = "Myrdsdb"
} -> null
- username                                = "raghavdb" -> null
- vpc_security_group_ids                  = [
  - "sg-03cc52f2e5e1b06eb",
] -> null
}
```

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

aws\_db\_instance.My-RDS: Destroying... [id=db-5MFTCL7JYNS3QWDBJB44EPRRY]

aws\_db\_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QWDBJB44EPRRY, 10s elapsed]

aws\_db\_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QWDBJB44EPRRY, 20s elapsed]

aws\_db\_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QWDBJB44EPRRY, 30s elapsed]

aws\_db\_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QWDBJB44EPRRY, 40s elapsed]

aws\_db\_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QWDBJB44EPRRY, 50s elapsed]

```
Plan: 0 to add, 0 to change, 1 to destroy.
aws_db_instance.My-RDS: Destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 10s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 20s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 30s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 40s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 50s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 1m0s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 1m10s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 1m20s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 1m30s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 1m40s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 1m50s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 2m0s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 2m10s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 2m20s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 2m30s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 2m40s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 2m50s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 3m0s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 3m10s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 3m20s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 3m30s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 3m40s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 3m50s elapsed]
aws_db_instance.My-RDS: Still destroying... [id=db-5MFTCL7JYNS3QMDBJBJ44EPRRY, 4m0s elapsed]
aws_db_instance.My-RDS: Destruction complete after 4m3s

Destroy complete! Resources: 1 destroyed.
PS C:\Users\anu39\Terraform-Script> |
```