

# SPCM LAB-10

Objective: Creating and AWS RDS instance using terraform

- Create a file rds.tf with the following contents

```
main.tf  rds.tf  ×
10th > rds.tf > resource "aws_db_instance" "my_RDS" > password
1  resource "aws_db_instance" "my_RDS" {
2      allocated_storage      = 10
3      db_name                 = "upes_db"
4      engine                 = "mysql"
5      engine_version         = "5.7"
6      instance_class         = "db.t2.micro"
7      username               = "admin"
8      password               = "admin1234"
9      skip_final_snapshot    = true
10     parameter_group_name    = "default.mysql5.7"
11
12 }
```

- Run terraform plan to check if the configurations align with your requirements

```
gauravbhandari@gauravs-Air-2 10th % 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
```

- Run terraform apply to create your resource

```
gauravbhandari@gauravs-Air-2 10th % terraform apply -auto-approve

Terraform used the selected providers to generate the following execution plan. Resource actions are
indicated with the following symbols:
+ create

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.
```

```
aws_db_instance.my_RDS: Still creating... [4m20s elapsed]
aws_db_instance.my_RDS: Still creating... [4m30s elapsed]
aws_db_instance.my_RDS: Still creating... [4m40s elapsed]
aws_db_instance.my_RDS: Still creating... [4m50s elapsed]
aws_db_instance.my_RDS: Creation complete after 4m56s [id=db-3VYESD7P524Z35H4NPJVL3VD4Q]

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

- Verify resource creation from AWS console.

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 Modify Actions Restore from S3 Create database

Filter by databases

DB identifier	Status	Role	Engine	Region & AZ	Size
<a href="#">terraform-20240226030311188900000001</a>	Available	Instance	MySQL Community	ap-south-1b	db.t2.micro

- After successful experimentation, run sql destroy to clean up the resources.

```
gauravbhandari@gauravs-Air-2 10th % terraform destroy -auto-approve
aws_db_instance.my_RDS: Refreshing state... [id=db-D2Y5MANAJ4DGMU3QIQ07C5KWSQ]

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 = "terraform-20240226030311188900000001.c5ouuqgashuw.ap-south-1.
    rds.amazonaws.com" -> null

aws_db_instance.my_RDS: Still destroying... [id=db-D2Y5MANAJ4DGMU3QIQ07C5KWSQ, 3m10s elapsed]
aws_db_instance.my_RDS: Still destroying... [id=db-D2Y5MANAJ4DGMU3QIQ07C5KWSQ, 3m20s elapsed]
aws_db_instance.my_RDS: Still destroying... [id=db-D2Y5MANAJ4DGMU3QIQ07C5KWSQ, 3m30s elapsed]
aws_db_instance.my_RDS: Still destroying... [id=db-D2Y5MANAJ4DGMU3QIQ07C5KWSQ, 3m40s elapsed]
aws_db_instance.my_RDS: Still destroying... [id=db-D2Y5MANAJ4DGMU3QIQ07C5KWSQ, 3m50s elapsed]
aws_db_instance.my_RDS: Still destroying... [id=db-D2Y5MANAJ4DGMU3QIQ07C5KWSQ, 4m0s elapsed]
aws_db_instance.my_RDS: Still destroying... [id=db-D2Y5MANAJ4DGMU3QIQ07C5KWSQ, 4m10s elapsed]
aws_db_instance.my_RDS: Destruction complete after 4m14s

Destroy complete! Resources: 1 destroyed.
```

