

# Module-12: Provisioning Infrastructure using Terraform Part-II

---

Demo Document - 1

edureka!

**edureka!**

© Brain4ce Education Solutions Pvt. Ltd.

## DEMO-1: Terraform State Subcommands

Here, we will demonstrate the list, pull, show, and rm subcommands.

1. To list all the resources in the current state

Syntax: terraform state list

```
ubuntu@ip-172-31-19-127:~/terra$ terraform state list
aws_instance.Terraform-instance1
aws_instance.Terraform-instance2
```

2. To pull and display the current state

Syntax: terraform state pull

```
ubuntu@ip-172-31-19-127:~/terra$ terraform state pull
{
  "version": 4,
  "terraform_version": "0.15.3",
  "serial": 28,
  "lineage": "608bc437-79ad-fa74-406b-ded9cceef2c24",
  "outputs": {},
  "resources": [
    {
      "mode": "managed",
      "type": "aws_instance",
      "name": "Terraform-instance1",
      "provider": "provider[\"aws_instance.Terraform-instance1\"]",
      "schema_version": 1,
      "data": {
        "ami": "ami-00399ec92321828f5",
        "arn": "arn:aws:ec2:us-east-2:763612999135:instance/i-03635ae7f16d53fdc",
        "associate_public_ip_address": true,
        "availability_zone": "us-east-2a",
        "cpu_core_count": 1,
        "cpu_threads_per_core": 1,
        "disable_api_termination": false,
        "ebs_optimized": false,
        "get_password_data": false,
        "hibernation": false,
        "id": "i-03635ae7f16d53fdc",
        "instance_initiated_shutdown_behavior": "stop",
        "instance_state": "running",
        "instance_type": "t2.micro",
        "ipv6_address_count": 0,
        "key_name": null,
        "monitoring": false,
        "network_interface_ids": [],
        "placement_group": null,
        "private_ip": "172.31.19.127",
        "public_ip": "54.154.154.154",
        "security_groups": ["sg-1a2b3c4d"],
        "subnet_id": "subnet-1a2b3c4d",
        "tags": {},
        "user_data": null,
        "vpc_security_group_ids": ["sg-1a2b3c4d"]
      }
    }
  ]
}
```

3. To show details about a particular resource in the state

Syntax: terraform state show <resource name from the list>

```
ubuntu@ip-172-31-19-127:~/terra$ terraform state show aws_instance.Terraform-instance2
# aws_instance.Terraform-instance2:
resource "aws_instance" "Terraform-instance2" {
  ami                        = "ami-00399ec92321828f5"
  arn                       = "arn:aws:ec2:us-east-2:763612999135:instance/i-03635ae7f16d53fdc"
  associate_public_ip_address = true
  availability_zone         = "us-east-2a"
  cpu_core_count            = 1
  cpu_threads_per_core      = 1
  disable_api_termination   = false
  ebs_optimized              = false
  get_password_data         = false
  hibernation                = false
  id                        = "i-03635ae7f16d53fdc"
  instance_initiated_shutdown_behavior = "stop"
  instance_state            = "running"
  instance_type             = "t2.micro"
  ipv6_address_count        = 0
  key_name                  = null
  monitoring                 = false
  network_interface_ids     = []
  placement_group           = null
  private_ip                = "172.31.19.127"
  public_ip                 = "54.154.154.154"
  security_groups            = ["sg-1a2b3c4d"]
  subnet_id                 = "subnet-1a2b3c4d"
  tags                      = {}
  user_data                 = null
  vpc_security_group_ids    = ["sg-1a2b3c4d"]
}
```

4. To remove an object from the state

Syntax: terraform state rm <name of the resource from the list>

```
ubuntu@ip-172-31-19-127:~/terra$ terraform state rm aws_instance.Terraform-instancel
Removed aws_instance.Terraform-instancel
Successfully removed 1 resource instance(s).
ubuntu@ip-172-31-19-127:~/terra$ terraform state list
aws_instance.Terraform-instance2
ubuntu@ip-172-31-19-127:~/terra$
```

edureka!