

Terraform Assignment – 3

```
1 provider "aws" {
2   region = "us-east-2"
3   access_key = "AKIA4MTWGW7C33ESVYP3"
4   secret_key = "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"
5 }
6
7 resource "aws_instance" "ohio_instance" {
8   ami      = "ami-0cb91c7de36eed2cb"
9   instance_type = "t2.micro"
10
11   tags = {
12     Name = "hello-ohio"
13   }
14 }
15
16 provider "aws" {
17
18   region = "us-east-1"
19   access_key = "AKIA4MTWGW7C33ESVYP3"
20   secret_key = "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"
21 }
22
23 resource "aws_instance" "virginia_instance" {
24   provider = aws.virginia
25   ami      = "ami-04b4f1a9cf54c11d0"
26   instance_type = "t2.micro"
27
28   tags = {
29     Name = "hello-virginia"
30   }
31 }
32
```

```
user@user:~/terra$ terraform apply
```

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

- + create

Terraform will perform the following actions:

```
# aws_instance.ohio_instance will be created
+ resource "aws_instance" "ohio_instance" {
  + ami                  = "ami-0cb91c7de36eed2cb"
  + arn                  = (known after apply)
  + associate_public_ip_address = (known after apply)
  + availability_zone     = (known after apply)
  + cpu_core_count       = (known after apply)
  + cpu_threads_per_core = (known after apply)
  + disable_api_stop     = (known after apply)
  + disable_api_termination = (known after apply)
  + ebs_optimized        = (known after apply)
  + enable_primary_ipv6   = (known after apply)
  + get_password_data     = false
  + host_id               = (known after apply)
  + host_resource_group_arn = (known after apply)
  + iam_instance_profile  = (known after apply)
  + id                    = (known after apply)
  + instance_initiated_shutdown_behavior = (known after apply)
  + instance_lifecycle    = (known after apply)
  + instance_state        = (known after apply)
  + instance_type         = "t2.micro"
  + ipv6_address_count    = (known after apply)
  + ipv6_addresses       = (known after apply)
  + key_name              = (known after apply)
  + monitoring            = (known after apply)
  + outpost_arn           = (known after apply)
  + password_data         = (known after apply)
  + placement_group       = (known after apply)
  + placement_partition_number = (known after apply)
  + primary_network_interface_id = (known after apply)
  + private_dns           = (known after apply)
  + private_ip            = (known after apply)
```

```

+ private_ip           = (known after apply)
+ public_dns           = (known after apply)
+ public_ip            = (known after apply)
+ secondary_private_ips = (known after apply)
+ security_groups       = (known after apply)
+ source_dest_check     = true
+ spot_instance_request_id = (known after apply)
+ subnet_id            = (known after apply)
+ tags                 = {
  + "Name" = "hello-virginia"
}
+ tags_all             = {
  + "Name" = "hello-virginia"
}
+ tenancy              = (known after apply)
+ user_data            = (known after apply)
+ user_data_base64     = (known after apply)
+ user_data_replace_on_change = false
+ vpc_security_group_ids = (known after apply)

+ capacity_reservation_specification (known after apply)

+ cpu_options (known after apply)

+ ebs_block_device (known after apply)

+ enclave_options (known after apply)

+ ephemeral_block_device (known after apply)

+ instance_market_options (known after apply)

+ maintenance_options (known after apply)

+ metadata_options (known after apply)

+ network_interface (known after apply)

+ private_dns_name_options (known after apply)

+ root_block_device (known after apply)
}

```

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.

Only 'yes' will be accepted to approve.

Enter a value: yes

```

+ "Name" = "hello-virginia"
}
+ tenancy = (known after apply)
+ user_data = (known after apply)
+ user_data_base64 = (known after apply)
+ user_data_replace_on_change = false
+ vpc_security_group_ids = (known after apply)

+ capacity_reservation_specification (known after apply)

+ cpu_options (known after apply)

+ ebs_block_device (known after apply)

+ enclave_options (known after apply)

+ ephemeral_block_device (known after apply)

+ instance_market_options (known after apply)

+ maintenance_options (known after apply)

+ metadata_options (known after apply)

+ network_interface (known after apply)

+ private_dns_name_options (known after apply)

+ root_block_device (known after apply)
}

```

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.

Only 'yes' will be accepted to approve.

Enter a value: yes

aws_instance.virginia_instance: Creating...

aws_instance.ohio_instance: Creating...

aws_instance.virginia_instance: Still creating... [10s elapsed]

aws_instance.ohio_instance: Still creating... [10s elapsed]

aws_instance.virginia_instance: Creation complete after 18s [id=i-04152cff77ede2c4e]

aws_instance.ohio_instance: Creation complete after 17s [id=i-0aae452f969827f8b]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

user@user:~/terra\$

The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with various AWS services like VPC, EFS, S3, CloudFormation, RDS, Lambda, Cloud9, AWS Glue, Athena, Simple Notification Service, Elastic Beanstalk, and Simple Queue Service. Below this, the 'Instances' page is displayed, showing a table with one instance named 'hello-ohio'. The instance is in the 'Running' state, has an ID of 'i-0aae452f969827f8b', and is a 't2.micro' instance. The console also shows a search bar and various filters for the instances.

[Alt+S]									
VPC EFS S3 CloudFormation RDS Lambda Cloud9 AWS Glue Athena Simple Notification Service Elastic Beanstalk Simple Queue Service									
United States (N. Virginia) rabeht147									
Instances (1) info									
Find Instance by attribute or tag (case-sensitive) All states									
Last updated less than a minute ago Connect Instance state Actions Launch instances									
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pul	
<input type="checkbox"/>	hello-virginia	i-04152cff77ede2c4e	Running	t2.micro	Initializing	View alarms +	us-east-1b	ec2	

```

user@user:~/terra$ terraform destroy
aws_instance.ohio_instance: Refreshing state... [id=i-0aae452f969827f8b]
aws_instance.virginia_instance: Refreshing state... [id=i-04152cff77ede2c4e]

Terraform used the selected providers to generate the following execution plan. Resources to be destroyed:
- destroy

Terraform will perform the following actions:

# aws_instance.ohio_instance will be destroyed
- resource "aws_instance" "ohio_instance" {
  - ami                               = "ami-0cb91c7de36eed2cb" -> null
  - arn                               = "arn:aws:ec2:us-east-2:851725170629:instance/i-0aae452f969827f8b" -> null
  - associate_public_ip_address      = true -> null
  - availability_zone                 = "us-east-2a" -> null
  - cpu_core_count                    = 1 -> null
  - cpu_threads_per_core              = 1 -> null
  - disable_api_stop                  = false -> null
  - disable_api_termination           = false -> null
  - ebs_optimized                     = false -> null
  - get_password_data                 = false -> null
  - hibernation                       = false -> null
  - id                               = "i-0aae452f969827f8b" -> null
  - instance_initiated_shutdown_behavior = "stop" -> null
  - instance_state                    = "running" -> null
  - instance_type                     = "t2.micro" -> null
  - ipv6_address_count                = 0 -> null
  - ipv6_addresses                    = [] -> null
  - monitoring                        = false -> null
  - placement_partition_number        = 0 -> null
  - primary_network_interface_id      = "eni-0d40dc8b79d947b81" -> null
  - private_dns                       = "ip-172-31-4-76.us-east-2.compute.internal" -> null
  - private_ip                       = "172.31.4.76" -> null
  - public_dns                        = "ec2-3-142-119-9.us-east-2.compute.amazonaws.com" -> null
  - public_ip                         = "3.142.119.9" -> null
  - secondary_private_ips              = [] -> null
  - security_groups                   = [
    - "default",
  ] -> null
  - source_dest_check                 = true -> null

```



```
user@user: ~/terra
}

- private_dns_name_options {
  - enable_resource_name_dns_a_record    = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type                        = "ip-name" -> null
}

- root_block_device {
  - delete_on_termination = true -> null
  - device_name           = "/dev/sda1" -> null
  - encrypted             = false -> null
  - iops                  = 3000 -> null
  - tags                  = {} -> null
  - tags_all              = {} -> null
  - throughput            = 125 -> null
  - volume_id             = "vol-08779dc92e9fb79c2" -> null
  - volume_size           = 8 -> null
  - volume_type           = "gp3" -> null
  # (1 unchanged attribute hidden)
}
}

Plan: 0 to add, 0 to change, 2 to destroy.

Do you really want to destroy all resources?
Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

aws_instance.ohio_instance: Destroying... [id=i-0aae452f969827f8b]
aws_instance.virginia_instance: Destroying... [id=i-04152cff77ede2c4e]
aws_instance.ohio_instance: Still destroying... [id=i-0aae452f969827f8b, 10s elapsed]
aws_instance.virginia_instance: Still destroying... [id=i-04152cff77ede2c4e, 10s elapsed]
aws_instance.ohio_instance: Still destroying... [id=i-0aae452f969827f8b, 20s elapsed]
aws_instance.virginia_instance: Still destroying... [id=i-04152cff77ede2c4e, 20s elapsed]
aws_instance.ohio_instance: Still destroying... [id=i-0aae452f969827f8b, 30s elapsed]
aws_instance.virginia_instance: Still destroying... [id=i-04152cff77ede2c4e, 30s elapsed]
aws_instance.ohio_instance: Still destroying... [id=i-0aae452f969827f8b, 40s elapsed]
aws_instance.virginia_instance: Still destroying... [id=i-04152cff77ede2c4e, 40s elapsed]
aws_instance.ohio_instance: Still destroying... [id=i-0aae452f969827f8b, 50s elapsed]
aws_instance.virginia_instance: Still destroying... [id=i-04152cff77ede2c4e, 50s elapsed]
aws_instance.ohio_instance: Still destroying... [id=i-0aae452f969827f8b, 1m0s elapsed]
aws_instance.virginia_instance: Still destroying... [id=i-04152cff77ede2c4e, 1m0s elapsed]
aws_instance.virginia_instance: Destruction complete after 1m5s
aws_instance.ohio_instance: Still destroying... [id=i-0aae452f969827f8b, 1m10s elapsed]
aws_instance.ohio_instance: Destruction complete after 1m15s

Destroy complete! Resources: 2 destroyed.
user@user:~/terra$
```