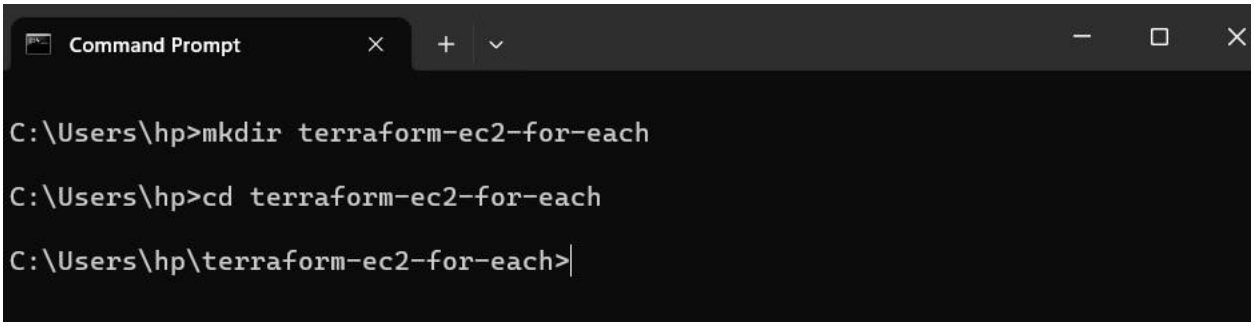
Lab-9

Creating Multiple EC2 Instances with for_ each in Terraform

Step 1: Create a Terraform Directory



main.tf

```
provider "aws" {
 region = "ap-south-1"
 Access_key ="AKIAV2D7UZ5ZGP6GB3P4"
 secret_key ="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
}
variable "instances" {
 description = "Map of EC2 instances with settings"
 default = {
 "instance1" = {
ami = "ami-0c55b159cbfafe1f0"
 instance_type = "t2.micro"
 },
 "instance2" = {
 ami = "ami-0123456789abcdef0"
 instance_type = "t2. micro "
 },
 "instance3" = {
 ami = "ami-9876543210fedcba0"
 instance_type = "t2. micro "
 }
 }
}
resource "aws_instance" "ec2_instances" {
 for_each = var.instances
 ami = var.instances[each.key].ami
 instance_type = var.instances[each.key].instance_type
 tags = {
 Name = "EC2-Instance-${each.key}"
 }
}
```

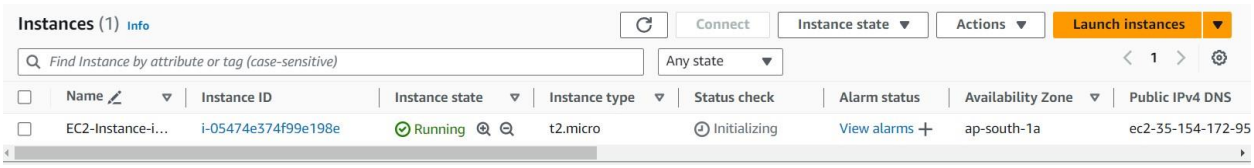
Step 2: Initialize and Apply

```
C:\Users\hp\terraform-ec2-for-each>terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.37.0...
- Installed hashicorp/aws v5.37.0 (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 reposito
ry
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 se
e
any changes that are required for your infrastructure. All Terraform command
s
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, ot
her
commands will detect it and remind you to do so if necessary.

C:\Users\hp\terraform-ec2-for-each>
```

```
C:\Users\hp\terraform-ec2-for-each>terraform apply

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_instance.ec2_instances["instance1"] will be created
  + resource "aws_instance" "ec2_instances" {
      + ami                                  = "ami-03f4878755434977f"
      + 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)
      + 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)
      + public_dns                           = (known after apply)
```

```
          + "Name" = "EC2-Instance-instance3"
        }
      + 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)
    }

Plan: 3 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.ec2_instances["instance3"]: Creating...
aws_instance.ec2_instances["instance2"]: Creating...
aws_instance.ec2_instances["instance1"]: Creating...
aws_instance.ec2_instances["instance1"]: Still creating... [10s elapsed]
aws_instance.ec2_instances["instance1"]: Still creating... [20s elapsed]
aws_instance.ec2_instances["instance1"]: Still creating... [30s elapsed]
aws_instance.ec2_instances["instance1"]: Still creating... [40s elapsed]
aws_instance.ec2_instances["instance1"]: Creation complete after 45s [id=i-05474e374f99e198e]
```

Step 3: Verify Instances in AWS Console



Step 4: Clean up

```
C:\Users\hp\terraform-ec2-for-each>terraform destroy
aws_instance.ec2_instances["instance1"]: Refreshing state... [id=i-05474e374f99e198e]

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_instance.ec2_instances["instance1"] will be destroyed
  - resource "aws_instance" "ec2_instances" {
      - ami                                  = "ami-03f4878755434977f" -> null
      - arn                                  = "arn:aws:ec2:ap-south-1:399699660658:instance/i-05474e374f99e198e" -> null
      - associate_public_ip_address          = true -> null
      - availability_zone                    = "ap-south-1a" -> 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-05474e374f99e198e" -> 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-0a3d6a38492cf97d7" -> null
      - private_dns                          = "ip-172-31-32-209.ap-south-1.compute.internal" -> null
      - private_ip                           = "172.31.32.209" -> null
      - public_dns                           = "ec2-35-154-172-95.ap-south-1.compute.amazonaws.com" -> null
      - public_ip                            = "35.154.172.95" -> null
      - secondary_private_ips                = [] -> null
      - security_groups                      = [
          - "default",
        ] -> null
      - source_dest_check                    = true -> null
```

```
      - root_block_device {
          - delete_on_termination = true -> null
          - device_name           = "/dev/sda1" -> null
          - encrypted             = false -> null
          - iops                  = 100 -> null
          - tags                  = {} -> null
          - throughput            = 0 -> null
          - volume_id             = "vol-0b50b237d9bc52ce2" -> null
          - volume_size           = 8 -> null
          - volume_type           = "gp2" -> null
        }
    }

Plan: 0 to add, 0 to change, 1 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.ec2_instances["instance1"]: Destroying... [id=i-05474e374f99e198e]
aws_instance.ec2_instances["instance1"]: Still destroying... [id=i-05474e374f99e198e, 10s elapsed]
aws_instance.ec2_instances["instance1"]: Still destroying... [id=i-05474e374f99e198e, 20s elapsed]
aws_instance.ec2_instances["instance1"]: Still destroying... [id=i-05474e374f99e198e, 30s elapsed]
aws_instance.ec2_instances["instance1"]: Destruction complete after 32s

Destroy complete! Resources: 1 destroyed.
```