

Question 6

Write a Terraform Script to create a sample infrastructure in the public cloud on AWS. Follow the diagram provided on BB. Upload the script to your Repo

1. Terraform and AWS CLI installation.

```
Command Prompt
Microsoft Windows [Version 10.0.19044.1387]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sreni>terraform version
Terraform v1.0.11
on windows_amd64

C:\Users\sreni>aws --version
aws-cli/2.4.1 Python/3.8.8 Windows/10 exe/AMD64 prompt/off

C:\Users\sreni>
```

2. Terraform Initialization

```
Command Prompt

C:\terra_demo_aws>terraform init

Initializing the backend...

Initializing provider plugins...
- Finding hashicorp/aws versions matching "~> 3.27"...
- Installing hashicorp/aws v3.67.0...
- Installed hashicorp/aws v3.67.0 (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.
```

3. Terraform format and validation

```
C:\> Command Prompt

C:\terra_demo_aws>terraform fmt

C:\terra_demo_aws>terraform validate
Success! The configuration is valid.

C:\terra_demo_aws>
```

4. Terraform apply

```
Command Prompt - terraform apply

C:\terra_demo_aws>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.app_server will be created
+ resource "aws_instance" "app_server" {
+   ami               = "ami-830c94e3"
+   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_termination = (known after apply)
+   ebs_optimized      = (known after apply)
+   get_password_data   = false
+   host_id            = (known after apply)
+   id                = (known after apply)
+   instance_initiated_shutdown_behavior = (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)
+   public_ip          = (known after apply)
+   secondary_private_ips = (known after apply)
+   security_groups     = (known after apply)
+   source_dest_check   = true
+   subnet_id          = (known after apply)
+   tags               = {
+     "Name" = "LVITExampleAppServerInstance"
+   }
+   tags_all           = {
+     "Name" = "LVITExampleAppServerInstance"
+   }
+   tenancy             = (known after apply)
+   user_data           = (known after apply)
+   user_data_base64    = (known after apply)
+   vpc_security_group_ids = (known after apply)

+ capacity_reservation_specification {
+   capacity_reservation_preference = (known after apply)

+   capacity_reservation_target {
+     capacity_reservation_id = (known after apply)
+   }
+ }

+ ebs_block_device {
+   delete_on_termination = (known after apply)
+   device_name           = (known after apply)
+   encrypted             = (known after apply)
+   iops                  = (known after apply)
+   kms_key_id            = (known after apply)
+   snapshot_id           = (known after apply)
+   tags                  = (known after apply)
+   throughput            = (known after apply)
+   volume_id             = (known after apply)
+   volume_size           = (known after apply)
+   volume_type           = (known after apply)
+ }
```

5. Create VM on AWS

```
Plan: 1 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.app_server: Creating...
aws_instance.app_server: Still creating... [10s elapsed]
aws_instance.app_server: Still creating... [20s elapsed]
aws_instance.app_server: Still creating... [30s elapsed]
aws_instance.app_server: Still creating... [40s elapsed]
aws_instance.app_server: Creation complete after 42s [id=i-0d4e556979c3e1a4b]

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

C:\terra_demo_aws>
```

6. Confirmation of VM on AWS

Instances (1/1) Info

Filter instances

Instance state: running X Clear filters

| <input checked="" type="checkbox"/> | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Av |
|-------------------------------------|-----------------|---------------------|----------------|---------------|--------------|--------------|----|
| <input checked="" type="checkbox"/> | LYITExampleA... | i-0d4e556979c3e1a4b | Running | t2.micro | Initializing | No alarms | eu |

Instance: i-0d4e556979c3e1a4b (LYITExampleAppServerInstance)

Details Security Networking Storage Status checks Monitoring Tags

▼ Instance summary Info

| | | |
|--------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| Instance ID i-0d4e556979c3e1a4b (LYITExampleAppServerInstance) | Public IPv4 address 3.250.29.136 open address | Private IPv4 addresses 172.31.34.85 |
| IPv6 address - | Instance state Running | Public IPv4 DNS ec2-3-250-29-136.eu-west-1.compute.amazonaws.com open address |
| Hostname type IP name: ip-172-31-34-85.eu-west-1.compute.internal | Private IP DNS name (IPv4 only) ip-172-31-34-85.eu-west-1.compute.internal | Answer private resource DNS name - |
| Instance type t2.micro | Elastic IP addresses - | VPC ID vpc-008c7f34d1bce809c |
| AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more | IAM Role - | Subnet ID subnet-0c732be4a405a126e |

▼ Instance details Info

| | | |
|------------------------------------------------------------------------------------|----------------------------------------------------------------------------|------------------------------------|
| Platform Amazon Linux (Inferred) | AMI ID ami-09ce2fc392a4c0fbc | Monitoring disabled |
| Platform details Linux/UNIX | AMI name amzn2-ami-kernel-5.10-hvm-2.0.20211103.1-x86_64-gp2 | Termination protection Disabled |
| Launch time Fri Nov 26 2021 12:45:28 GMT+0000 (Greenwich Mean Time) (2 minutes) | AMI location amazon/amzn2-ami-kernel-5.10-hvm-2.0.20211103.1-x86_64-gp2 | Lifecycle normal |
| Stop-hibernate behavior disabled | AMI Launch index 0 | Key pair name - |

7. Changing ami

```
Command Prompt

C:\terra_demo_aws>terraform validate
Success! The Configuration is valid.

C:\terra_demo_aws>terraform plan
aws_instance.app_server: Refreshing state... [id=i-0d4e556979c3e1a4b]

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

Terraform will perform the following actions:

# aws_instance.app_server must be replaced
/+ resource "aws_instance" "app_server" {
  ~ ami                  = "ami-09ce2fc392a4c0fbc" -> "ami-0de648cb0cb33b332" # forces replacement
  ~ arn                  = "arn:aws:ec2:eu-west-1:785521143660:instance/i-0d4e556979c3e1a4b" -> (known after apply)
  ~ associate_public_ip_address = true -> (known after apply)
  ~ availability_zone      = "eu-west-1a" -> (known after apply)
  ~ cpu_core_count        = 1 -> (known after apply)
  ~ cpu_threads_per_core   = 1 -> (known after apply)
  ~ disable_api_termination = false -> (known after apply)
  ~ ebs_optimized          = false -> (known after apply)
  ~ hibernation            = false -> null
  + host_id                = (known after apply)
  ~ id                     = "i-0d4e556979c3e1a4b" -> (known after apply)
  ~ instance_initiated_shutdown_behavior = "stop" -> (known after apply)
  ~ instance_state         = "running" -> (known after apply)
  ~ ipv6_address_count      = 0 -> (known after apply)
  ~ ipv6_addresses         = [] -> (known after apply)
  ~ key_name                = (known after apply)
  ~ monitoring              = false -> (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 = "eni-04f894fe1c46be089" -> (known after apply)
  ~ private_dns             = "ip-172-31-34-85.eu-west-1.compute.internal" -> (known after apply)
  ~ private_ip              = "172.31.34.85" -> (known after apply)
  ~ public_dns              = "ec2-3-250-29-136.eu-west-1.compute.amazonaws.com" -> (known after apply)
  ~ public_ip               = "3.250.29.136" -> (known after apply)
  ~ secondary_private_ips    = [] -> (known after apply)
  ~ security_groups         = [
    - "default",
  ] -> (known after apply)
  ~ subnet_id              = "subnet-0c732be4a405a126e" -> (known after apply)
  ~ tags                   = {
    "Name" = "LYITExampleAppServerInstance"
  }
  ~ tenancy                = "default" -> (known after apply)
  + user_data               = (known after apply)
  + user_data_base64        = (known after apply)
  ~ vpc_security_group_ids = [
    - "sg-0084ee2e22b653424",
  ] -> (known after apply)
  # (4 unchanged attributes hidden)

  ~ capacity_reservation_specification {
    ~ capacity_reservation_preference = "open" -> (known after apply)

    + capacity_reservation_target {
      + capacity_reservation_id = (known after apply)
    }
  }

  - credit_specification {
    - cpu_credits = "standard" -> null
  }
}
```

8. Destroying and creating new ami

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

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

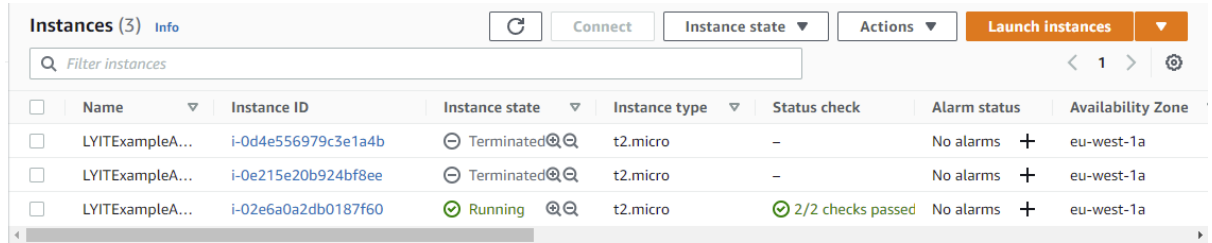
  Enter a value: yes

aws_instance.app_server: Destroying... [id=i-0e215e20b924bf8ee]
aws_instance.app_server: Still destroying... [id=i-0e215e20b924bf8ee, 10s elapsed]
aws_instance.app_server: Still destroying... [id=i-0e215e20b924bf8ee, 20s elapsed]
aws_instance.app_server: Destruction complete after 30s
aws_instance.app_server: Creating...
aws_instance.app_server: Still creating... [10s elapsed]
aws_instance.app_server: Still creating... [20s elapsed]
aws_instance.app_server: Still creating... [30s elapsed]
aws_instance.app_server: Still creating... [40s elapsed]
aws_instance.app_server: Creation complete after 43s [id=i-02e6a0a2db0187f60]

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

C:\terra_demo_aws>
```

9. AWS terminate old and run new ami



| | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|--------------------------|-----------------|---------------------|----------------|---------------|-------------------|--------------|-------------------|
| <input type="checkbox"/> | LYITExampleA... | i-0d4e556979c3e1a4b | Terminated | t2.micro | - | No alarms | eu-west-1a |
| <input type="checkbox"/> | LYITExampleA... | i-0e215e20b924bf8ee | Terminated | t2.micro | - | No alarms | eu-west-1a |
| <input type="checkbox"/> | LYITExampleA... | i-02e6a0a2db0187f60 | Running | t2.micro | 2/2 checks passed | No alarms | eu-west-1a |

10. Changes to var file

```
Plan: 0 to add, 1 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.

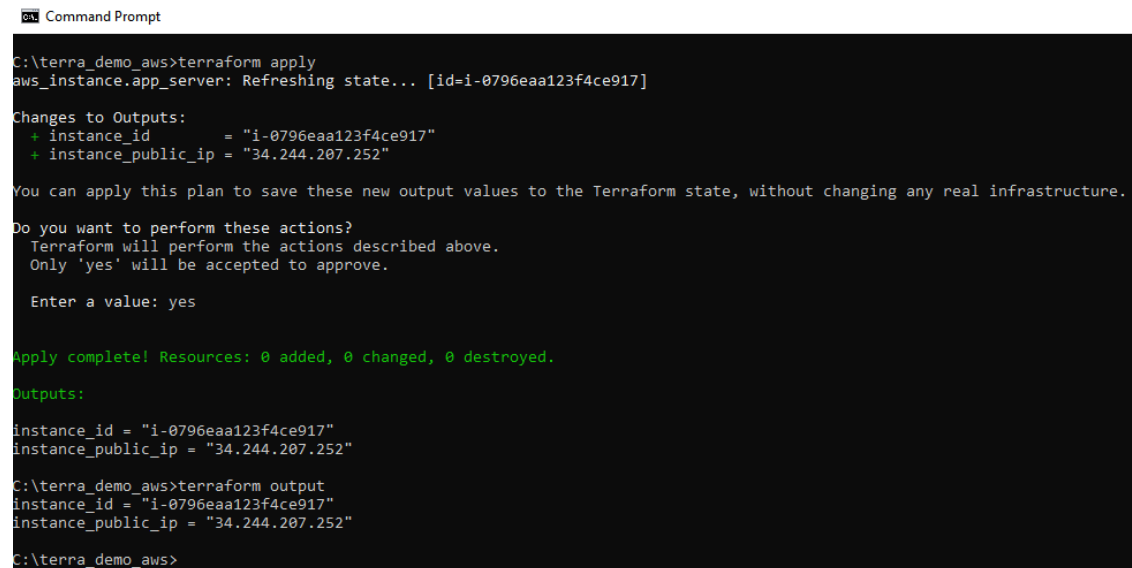
  Enter a value: yes

aws_instance.app_server: Modifying... [id=i-0796eaa123f4ce917]
aws_instance.app_server: Modifications complete after 1s [id=i-0796eaa123f4ce917]

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

C:\terra_demo_aws>
```

11. Outputs:



```
Command Prompt

C:\terra_demo_aws>terraform apply
aws_instance.app_server: Refreshing state... [id=i-0796eaa123f4ce917]

Changes to Outputs:
  + instance_id      = "i-0796eaa123f4ce917"
  + instance_public_ip = "34.244.207.252"

You can apply this plan to save these new output values to the Terraform state, without changing any real infrastructure.

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

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

Outputs:
instance_id = "i-0796eaa123f4ce917"
instance_public_ip = "34.244.207.252"

C:\terra_demo_aws>terraform output
instance_id = "i-0796eaa123f4ce917"
instance_public_ip = "34.244.207.252"

C:\terra_demo_aws>
```

12. Destroying Instance

Command Prompt - terraform destroy

```
C:\terra_demo_aws>terraform destroy
aws_instance.app_server: Refreshing state... [id=i-02e6a0a2db0187f60]

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.app_server will be destroyed
- resource "aws_instance" "app_server" {
  - ami                  = "ami-09ce2fc392a4c0fbc" -> null
  - arn                  = "arn:aws:ec2:eu-west-1:785521143660:instance/i-02e6a0a2db0187f60" -> null
  - associate_public_ip_address = true -> null
  - availability_zone     = "eu-west-1a" -> null
  - cpu_core_count        = 1 -> null
  - cpu_threads_per_core   = 1 -> null
  - disable_api_termination = false -> null
  - ebs_optimized          = false -> null
  - get_password_data      = false -> null
  - hibernation            = false -> null
  - id                    = "i-02e6a0a2db0187f60" -> 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
  - primary_network_interface_id = "eni-01ca028fcc55c2edb" -> null
  - private_dns             = "ip-172-31-45-32.eu-west-1.compute.internal" -> null
  - private_ip             = "172.31.45.32" -> null
  - public_dns             = "ec2-54-247-30-214.eu-west-1.compute.amazonaws.com" -> null
  - public_ip              = "54.247.30.214" -> null
  - secondary_private_ips    = [] -> null
  - security_groups         = [
    - "default",
  ] -> null
  - source_dest_check       = true -> null
  - subnet_id              = "subnet-0c732be4a405a126e" -> null
  - tags                   = {
    - "Name" = "LYITExampleAppServerInstance"
  } -> null
  - tags_all               = {
    - "Name" = "LYITExampleAppServerInstance"
  } -> null
  - tenancy                 = "default" -> null
  - vpc_security_group_ids = [
    - "sg-0084ee2e22b653424",
  ] -> 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.app_server: Destroying... [id=i-02e6a0a2db0187f60]
aws_instance.app_server: Still destroying... [id=i-02e6a0a2db0187f60, 10s elapsed]
aws_instance.app_server: Still destroying... [id=i-02e6a0a2db0187f60, 20s elapsed]
aws_instance.app_server: Still destroying... [id=i-02e6a0a2db0187f60, 30s elapsed]
▼ aws_instance.app_server: Still destroying... [id=i-02e6a0a2db0187f60, 40s elapsed]
aws_instance.app_server: Destruction complete after 40s

Destroy complete! Resources: 1 destroyed.

C:\terra_demo_aws>

▼

Instances (3) Info

🔄

Connect

Instance state ▼

Actions ▼

Launch instances ▼

🔍 Filter instances

< 1 > ⚙️

| <input type="checkbox"/> | Name ▼ | Instance ID | Instance state ▼ | Instance type ▼ | Status check | Alarm status | Availability Zone |
|--------------------------|-----------------|---------------------|------------------|-----------------|--------------|--------------|-------------------|
| <input type="checkbox"/> | LYITExampleA... | i-0d4e556979c3e1a4b | Terminated🔍🔍 | t2.micro | - | No alarms + | eu-west-1a |
| <input type="checkbox"/> | LYITExampleA... | i-0e215e20b924bf8ee | Terminated🔍🔍 | t2.micro | - | No alarms + | eu-west-1a |
| <input type="checkbox"/> | LYITExampleA... | i-02e6a0a2db0187f60 | Terminated🔍🔍 | t2.micro | - | No alarms + | eu-west-1a |