TERRAFORM RUNBOOK to create Autoscaling group

terraform {

required\_providers {

aws = {

source = "hashicorp/aws"

version = "~> 4.16"

}

}

required\_version = ">= 1.2.0"

}

provider "aws" {

region = "ap-south-1"

}

# Security Group for EC2 Instances

resource "aws\_security\_group" "web\_sg" {

name = "web\_sg"

description = "Allow HTTP and SSH traffic"

vpc\_id = "vpc-0215cff6cab57ccb5"

ingress {

from\_port = 80

to\_port = 80

protocol = "tcp"

cidr\_blocks = ["0.0.0.0/0"]

}

ingress {

from\_port = 22

to\_port = 22

protocol = "tcp"

cidr\_blocks = ["0.0.0.0/0"]

}

egress {

from\_port = 0

to\_port = 0

protocol = "-1"

cidr\_blocks = ["0.0.0.0/0"]

}

}

# Launch Template for EC2 Instances

resource "aws\_launch\_template" "web\_launch\_template" {

name\_prefix = "web-instance-template"

image\_id = "ami-0522ab6e1ddcc7055" # Replace with your desired AMI

instance\_type = "t2.micro"

key\_name = "aws-keypair"

network\_interfaces {

associate\_public\_ip\_address = true

security\_groups = [aws\_security\_group.web\_sg.id]

}

}

# Auto Scaling Group

resource "aws\_autoscaling\_group" "web\_asg" {

desired\_capacity = 1

max\_size = 3

min\_size = 1

vpc\_zone\_identifier = ["subnet-056c6ccf114ab2079"] # Your subnet ID

launch\_template {

id = aws\_launch\_template.web\_launch\_template.id

version = "$Latest"

}

health\_check\_type = "EC2"

health\_check\_grace\_period = 300

tag {

key = "Name"

value = "WebServer"

propagate\_at\_launch = true

}

}

# CloudWatch Alarms for Scaling

resource "aws\_cloudwatch\_metric\_alarm" "scale\_up\_alarm" {

alarm\_name = "cpu\_too\_high"

comparison\_operator = "GreaterThanThreshold"

evaluation\_periods = 2

metric\_name = "CPUUtilization"

namespace = "AWS/EC2"

period = 120

statistic = "Average"

threshold = 70

alarm\_actions = [aws\_autoscaling\_policy.scale\_up\_policy.arn]

dimensions = {

AutoScalingGroupName = aws\_autoscaling\_group.web\_asg.name

}

}

resource "aws\_cloudwatch\_metric\_alarm" "scale\_down\_alarm" {

alarm\_name = "cpu\_too\_low"

comparison\_operator = "LessThanThreshold"

evaluation\_periods = 2

metric\_name = "CPUUtilization"

namespace = "AWS/EC2"

period = 120

statistic = "Average"

threshold = 20

alarm\_actions = [aws\_autoscaling\_policy.scale\_down\_policy.arn]

dimensions = {

AutoScalingGroupName = aws\_autoscaling\_group.web\_asg.name

}

}

# Scaling Policy

resource "aws\_autoscaling\_policy" "scale\_up\_policy" {

name = "scale-up"

scaling\_adjustment = 1

adjustment\_type = "ChangeInCapacity"

cooldown = 300

autoscaling\_group\_name = aws\_autoscaling\_group.web\_asg.name

}

resource "aws\_autoscaling\_policy" "scale\_down\_policy" {

name = "scale-down"

scaling\_adjustment = -1

adjustment\_type = "ChangeInCapacity"

cooldown = 300

autoscaling\_group\_name = aws\_autoscaling\_group.web\_asg.name

}