Hello Guys ,  
  
Using Terraform , now we gonna create a master slave configuration  
  
In master node , we gonna install Jenkins and java  
  
In slave node , we gonna create java , docker , maven  
  
only the above things we gonna create using terraform

1. For that Create an ec2 intstance , t2.micro

2. create a IAM role , Give ADMIN ACCESS if needed

3. Create a security group ,   
 select ALL TRAFFIC or ALL TCP , SSH  
4. Login to the server through SSH   
  
update the packages by  
  
**apt-get update**

5. create a dir cd /opt/terraform

And install terraform from official website

<https://developer.hashicorp.com/terraform/install>

wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg

echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb\_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list

sudo apt update && sudo apt install terraform

6. Create a main.tf file using vi editor

vi main.tf  
  
  
AND paste the below script , if possible modify it according to your needs  
Please go throught script and learn it what the script is doing  
  
**Please do change vpc id , subnet id , ami**  
  
  
  
provider "aws" {

  region = "ap-south-1"

}

# Use your existing VPC and Subnet IDs

variable "vpc\_id" {

  description = "The ID of the VPC"

  default     = "vpc-0c0a81b0f23a0bbef"  # Replace with your VPC ID

}

variable "subnet\_id" {

  description = "The ID of the subnet"

  default     = "subnet-00fd9733fafc4c1ba"  # Replace with your Subnet ID

}

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

  vpc\_id = var.vpc\_id

  ingress {

    from\_port   = 22

    to\_port     = 22

    protocol    = "tcp"

    cidr\_blocks = ["0.0.0.0/0"]

  }

  ingress {

    from\_port   = 8080

    to\_port     = 8080

    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"]

  }

}

resource "aws\_instance" "jenkins\_master" {

  ami           = "ami-0f58b397bc5c1f2e8"  # Amazon Ubuntu AMI

  instance\_type = "t3.medium"

  subnet\_id     = var.subnet\_id

  associate\_public\_ip\_address = true  # Assign a public IP to the instance

  vpc\_security\_group\_ids = [aws\_security\_group.jenkins\_sg.id]  # Use vpc\_security\_group\_ids instead of security\_group\_ids

  key\_name      = "mumadvento"

  tags = {

    Name = "Jenkins-Master"

  }

  user\_data = <<-EOF

                #!/bin/bash

                # Update package list

                sudo apt-get update

                # Install Java (OpenJDK 11)

                sudo apt-get install -y openjdk-11-jdk

                # Add Jenkins repository and key

                curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee \

                /usr/share/keyrings/jenkins-keyring.asc > /dev/null

                echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian-stable binary/" | sudo tee \

                /etc/apt/sources.list.d/jenkins.list > /dev/null

                # Update package list again

                sudo apt-get update

                # Install Jenkins

                sudo apt-get install -y jenkins

                # Start Jenkins service

                sudo systemctl start jenkins

                sudo systemctl enable jenkins

                EOF

}

resource "aws\_instance" "jenkins\_slave" {

  ami           = "ami-0f58b397bc5c1f2e8"  # Amazon Ubuntu AMI

  instance\_type = "t3.medium"

  subnet\_id     = var.subnet\_id

  associate\_public\_ip\_address = true  # Assign a public IP to the instance

  vpc\_security\_group\_ids = [aws\_security\_group.jenkins\_sg.id]  # Use vpc\_security\_group\_ids instead of security\_group\_ids

  key\_name      = "mumadvento"

  tags = {

    Name = "Jenkins-Slave"

  }

  user\_data = <<-EOF

              #!/bin/bash

              # Update package list

              sudo apt-get update

              # Install Java (OpenJDK 11)

              sudo apt-get install -y openjdk-11-jdk

              # Install Docker

              sudo apt-get install -y docker.io

              sudo systemctl start docker

              sudo systemctl enable docker

              # Install Maven

              sudo apt-get install -y maven

              EOF

}

output "jenkins\_master\_public\_ip" {

  value = aws\_instance.jenkins\_master.public\_ip

}

This script will create a 2 instance   
  
MASTER and SLAVE   
  
In master , this script will install and start Jenkins and java  
  
In slave machine , this script will install and start java , docker , maven

We need to manually configure the setup of master and slave in Jenkins GUI .  
If you guys need the whole script to install and configure the whole setup of master and slave , then please do ping me in linkedin . I will give you the whole script .