Write a Terraform/Cloud Formation template for the LAMP stack in Question 2

Create a file **creds.tf**

```
# AWS Config

variable "aws_access_key" {
default = "<AWS Key here>"
}

variable "aws_secret_key" {
default = "<Secret Key here>"
}

variable "aws_region" {
default = "<Region for Launching Instance>"
}
```

Create another file **my-instance.tf**

```
provider "aws" {
access_key = "${var.aws_access_key}"
secret_key = "${var.aws_secret_key}"
region = "${var.aws_region}"
}
resource "aws_security_group" "" {
  name = ""
  description = "Test security group."
  vpc_id = ""
}


    security_group_id = "${aws_security_group.mysecuritygroup.id}"


resource "aws_security_group_rule" "ssh_ingress_access" {
  security_group_id = "<ID of the security Group >"
  cidr_blocks = [ "0.0.0.0/0" ]
}

resource "aws_security_group_rule" "egress_access" {
  security_group_id = "<ID of the security Group >"
  cidr_blocks = [ "0.0.0.0/0" ]
}

resource "aws_instance" "<name>" {
  subnet_id = ""
```

```
  instance_type = ""
  vpc_security_group_ids = [ ]
  associate_public_ip_address = false
  tags {
    Name = "<Name of the instance>"
  }

  ami = "ami-cb2305a1"
  availability_zone = "<In which zone the instance is to be launched>"

command = "sleep 120; ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -u ubuntu
--private-key ./<name of pem key with .pem extension> -i '${aws_instance.<name>.public_ip },'
<name of yml>"


}
```

Example ::

```
resource "aws_security_group" "test_security_group" {
  name = "dummy-sg"

  description = "Test security group."

  # This is fake VPC ID, you should put real one to make this configuration working
  vpc_id = "vpc-11111111"
}

resource "aws_security_group_rule" "ssh_ingress_access" {
  type = "ingress"
  from_port = 22
  to_port = 22
  protocol = "tcp"
  cidr_blocks = [ "0.0.0.0/0" ]
  security_group_id = "${aws_security_group.test_security_group.id}"
}

resource "aws_security_group_rule" "egress_access" {
  type = "egress"
  from_port = 0
  to_port = 65535
  protocol = "tcp"
  cidr_blocks = [ "0.0.0.0/0" ]
```

```
  security_group_id = "${aws_security_group.test_security_group.id}"
}

resource "aws_instance" "w1_instance" {
  instance_type = "t2.nano"
  vpc_security_group_ids = [ "${aws_security_group.test_security_group.id}" ]
  associate_public_ip_address = true
  user_data = "${file("../../shared/user-data.txt")}"
  tags {
    Name = "myinstance"
  }

  ami = "ami-cb2305a1"
  availability_zone = "us-east-1c"

  # This is fake VPC subnet ID, please put real one to make this config working
  subnet_id = "subnet-1111111"

command = "sleep 120; ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -u ubuntu
--private-key ./<test.pem> -i '${aws_instance.<name>.public_ip },'LAMP/main.yml"

}
```