Laboratório 15 – Módulos

# Cenário

Nesse laboratório iremos ver como utilizar módulos para simplificar o desenvolvimento e manutenção do código.

# Objetivos:

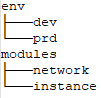
Nesse laboratório, você irá:

* Atividade 01: Criar estrutura de diretórios
* Atividade 02: Criar módulo network
* Atividade 03: Criar módulo instance
* Atividade 04: Provisionar ambiente de dev
* Atividade 05: Provisionar ambiente de prd
* Atividade 06: Removendo recursos

# Instruções

## Atividade 01: Criar estrutura de diretórios

1. Crie um diretório chamado lab15 dentro do diretório laboratórios.
2. Dentro de lab15 crie a seguinte estrutura de diretórios.



## Atividade 02: Criar módulo network

1. Dentro do diretório modules/network crie os seguintes arquivos.

* aws\_subnet.tf
* aws\_vpc.tf
* outputs.tf
* variables.tf

1. Adicione o seguinte conteúdo no arquivo aws\_subnet.tf

resource "aws\_subnet" "my\_subnet" {

vpc\_id = aws\_vpc.my\_vpc.id

cidr\_block = "172.16.10.0/24"

availability\_zone = "us-east-1a"

tags = {

"Name" = "${var.env}-demo-terraform"

"Environment" = "MyEnv"

}

}

1. Adicione o seguinte conteúdo no arquivo aws\_vpc.tf

resource "aws\_vpc" "my\_vpc" {

cidr\_block = "172.16.0.0/16"

tags = {

"Name" = "${var.env}-demo-terraform"

"Environment" = var.env

}

}

1. Adicione o seguinte conteúdo no arquivo outputs.tf

output "subnet\_id" {

value = aws\_subnet.my\_subnet.id

}

1. Adicione o seguinte conteúdo no arquivo variables.tf

variable "env" {

description = "Ambiente que será provisionado"

type = string

}

## Atividade 03: Criar módulo instance

1. Dentro do diretório modules/network crie os seguintes arquivos.

* aws\_instance.tf
* aws\_network\_instance.tf
* variables.tf

1. Adicione o seguinte conteúdo no arquivo aws\_instance.tf

resource "aws\_instance" "foo" {

ami = "ami-052efd3df9dad4825"

instance\_type = "t2.micro"

network\_interface {

network\_interface\_id = aws\_network\_interface.foo.id

device\_index = 0

}

tags = {

"Name" = "${var.env}-${var.ec2\_name}"

"Environment" = var.env

}

}

1. Adicione o seguinte conteúdo no arquivo aws\_network\_instance.tf

resource "aws\_network\_interface" "foo" {

subnet\_id = var.subnet\_id

private\_ips = ["172.16.10.100"]

tags = {

"Name" = "${var.env}-${var.ec2\_name}"

"Environment" = var.env

}

}

1. Adicione o seguinte conteúdo no arquivo variables.tf

variable "ec2\_name" {

description = "Nome da instância do EC2"

type = string

}

variable "subnet\_id" {

type = string

description = "Subnet ID to network interface"

}

variable "env" {

description = "Ambiente que será provisionado"

type = string

}

## Atividade 04: Provisionar ambiente de dev

1. Dentro do diretório env/dev crie os seguintes arquivos.

* module.tf
* provider.tf
* terraform.tf
* terraform.tfvars
* variables.tf

1. Adicione o seguinte conteúdo no arquivo module.tf

module "network" {

source = "../../modules/network"

env = var.env

}

module "instance" {

source = "../../modules/instance"

ec2\_name = "demo-terraform"

subnet\_id = module.network.subnet\_id

env = var.env

}

1. Adicione o seguinte conteúdo no arquivo provider.tf

provider "aws" {

region = "us-east-1"

access\_key = "my-access-key"

secret\_key = "my-secret-key"

}

1. Acesse o arquivo provider.tf e adicione o access key e secret key do seu usuário.
2. Adicione o seguinte conteúdo no arquivo terraform.tf

terraform {

required\_providers {

aws = {

source = "hashicorp/aws"

version = "4.30.0"

}

}

}

1. Adicione o seguinte conteúdo no arquivo terraform.tfvars

env="dev"

1. Adicione o seguinte conteúdo no arquivo variables.tf

variable "env" {

type = string

description = "Environment name"

}

1. Abra o terminal.
2. Execute o comando cd .\env\dev\
3. Execute o comando terraform init.
4. Execute o comando terraform apply --auto-approve.
5. Visualize os recursos provisionados.

## Atividade 05: Provisionar embiente de prd

1. Dentro do diretório env/prd crie os seguintes arquivos.

* module.tf
* provider.tf
* terraform.tf
* terraform.tfvars
* variables.tf

1. Adicione o seguinte conteúdo no arquivo module.tf

module "network" {

source = "../../modules/network"

env = var.env

}

module "instance" {

source = "../../modules/instance"

ec2\_name = "demo-terraform"

subnet\_id = module.network.subnet\_id

env = var.env

}

1. Adicione o seguinte conteúdo no arquivo provider.tf

provider "aws" {

region = "us-east-1"

access\_key = "my-access-key"

secret\_key = "my-secret-key"

}

1. Acesse o arquivo provider.tf e adicione o access key e secret key do seu usuário.
2. Adicione o seguinte conteúdo no arquivo terraform.tf

terraform {

required\_providers {

aws = {

source = "hashicorp/aws"

version = "4.30.0"

}

}

}

1. Adicione o seguinte conteúdo no arquivo terraform.tfvars

env="prd"

1. Adicione o seguinte conteúdo no arquivo variables.tf

variable "env" {

type = string

description = "Environment name"

}

1. Execute o comando cd ..\prd\
2. Execute o comando terraform init.
3. Execute o comando terraform apply --auto-approve.
4. Visualize os recursos provisionados.

## Atividade 06: Removendo recursos

1. Execute o comando terraform destroy --auto-approve.
2. Execute o comando cd ..\dev\
3. Execute o comando terraform destroy --auto-approve.