**Questão 1 - Quem criou o terraform?**

Microsoft

Google

HashiCorp

Docker

AWS

**Questão 2 - Qual o objetivo do terraform?**

API Rest para provisionamento de recursos na AWS.

Aplicação web para provisionamento de recursos na AWS.

Framework que abstrai o desenvolvimento de código.

Gerenciar e provisionar recursos.

**Questão 3 - O terraform é um executável (binário) e não precisamos rodar um wizard para fazer a instalação.**

Escolha uma opção:

Verdadeiro

Falso

**Questão 4 - Qual é o fluxo de trabalho do terraform para provisionar a infraestrutura?**

Plan, Apply e Init

Apply, Plan e Init

Init, Plan e Apply

Init, Plan e Deploy

Init, Plan e Build

**Questão 5 - Qual configuração utilizamos quando queremos reutilizar um conjunto de arquivo .tf?**

Dynamic block

Module

Resource

Datasource

Workspace

**Questão 6 - O precondition é utilizado em que momento?**

Quando é informado um valor para a variável.

Depois que for realizado o provisionamento.

No momento que o usuário quiser.

Antes que o terraform analise o recurso que será criado.

Depois que o terraform analise o recurso que será criado.

**Questão 7 - O precondition e o postcondition são configurados em qual sub-group?**

Ingress

lifecycle

tags

parameters

egress

**Questão 8 - Não temos como reutilizar o código contido no diretório de trabalho para vários ambientes.**

**Escolha uma opção:**

Verdadeiro

Falso

**Questão 9 - Qual variável padrão que podemos utilizar para trabalhar com o nome do workspace?**

var.workspace.name

${var.workspace}

var.workspace

terraform.workspace

terraform.space

**Questão 10 - Quando temos sub-grupo que se repetem dentro do bloco resource podemos utilizar o dynamic block para ter somente um sub-grupo e não vários.**

**Escolha uma opção:**

Verdadeiro

Falso

**Questão 11 - Junto com o dynamic block utilizamos o for\_each para que seja possível utilizar uma lista contendo os dados utilizados para configurar o sub-grupo.**

**Escolha uma opção:**

Verdadeiro

Falso

**Questão 12 - Onde encontro detalhes de como importar um recurso já existente.**

Não é possível importar um recurso já existente.

No registry do terraform.

Na documentação do provider.

No comando terraform import -help.

Na documentação do terraform.

**Questão 13 - O terraform permite que possamos criar nossos próprios provider caso um software/solução que trabalhamos não o possua.**

**Escolha uma opção:**

Verdadeiro

Falso

**Questão 14 - Qual forma de autenticação mais indicada para quando temos várias contas da AWS?**

Credências do usuário

Private Key

IAM Instance Profile

Certificado digital

Assume Role

**Questão 15 - Qual o nome do bloco usado para exibir informações de recursos provisionados?**

Module

Local

Terraform

Variable

Output

**Questão 16 - Qual opção podemos utilizar dentro do bloco output para documentar a informação que será exibida?**

default

description

validation

type

sensitive

**Questão 17 - Para não perder as credenciais do usuário devemos escrever dentro do bloco provider e subir para o repositório, com isso todos os colaboradores poderão usar a mesma credencial.**

**Escolha uma opção:**

Verdadeiro

Falso

**Questão 18 - Onde devemos executar o comando terraform init?**

No diretório modules

No diretório onde instalamos o terraform

No diretório onde contém os arquivos de configuração

No diretório onde o sistema operacional foi instalado

Em qualquer diretório

**Questão 19 - O que faz esse bloco?**

**resource "aws\_s3\_bucket" "mybucket" {**

**bucket = "mybucket"**

**tags = {**

**"environment" = "dev"**

**}**

**}**

Indica um módulo que cria um bucket do S3

Declara uma variável

Busca informação de um Bucket do S3

Cria um Bucket do S3

Cria um RDS

**Questão 20 - Como preenchemos o value de um output caso seja necessário informar o nome de todos os bucket do S3 provisionados com count?**

aws\_s3\_bucket.mybucket.0.name

aws\_s3\_bucket.mybucket.[index].name

aws\_s3\_bucket.mybucket.[0].name

aws\_s3\_bucket.mybucket.index.name

aws\_s3\_bucket.mybucket.\*.name