**Question 1Correct**

Matthew wants to set a Terraform variable using the environment variables.

What is the right format of environment variable name that needs to be defined?

**VAR\_TF\_NAME**

**TF\_VAR**

**TF\_VAR\_ENV\_name**

**Your answer is correct**

**TF\_VAR\_name**

**Overall explanation**

The environment variables must be in the format **TF\_VAR\_name** and this will be checked last for a value.

<https://www.terraform.io/docs/commands/environment-variables.html>

Few examples were variables are set using this use-case:

* export TF\_VAR\_region=us-west-1
* export TF\_VAR\_ami=ami-049d8641
* export TF\_VAR\_alist='[1,2,3]'

**Question 2Correct**

If John runs the terraform init command 10 times in a specific terraform project directory, can it lead to any corruption or modification of terraform configuration files?

**Yes**

**Your answer is correct**

**No**

**Question 3Correct**

Is it recommended to use Terraform Provisioners extensively, or should it be used minimally?

**Extensively**

**Your answer is correct**

**Minimally**

**Overall explanation**

Provisioners are the Last Resort. Make sure to always impart changes as part of the AMI (Image) itself rather than running provisioners for all the new servers.

<https://www.terraform.io/docs/provisioners/index.html>

**Question 4Correct**

Small Corp has 3 environments named Dev, Stage and QA.

Since they have a small team, they want to maintain a single configuration file but with a different terraform.tfstate file for each environment.

Which Terraform feature can be used?

**Modules**

**Backends**

**Functions**

**Your answer is correct**

**Workspaces**

**Resources**

**Overall explanation**

Terraform workspaces allow you to isolate your infrastructure into separate environments. Each workspace has its own state file.

**Question 5Correct**

James has set the following environment variable:

TF\_LOG\_PATH=./terraform-debug.log

However, even after running multiple Terraform operations, the logs are not stored in the terraform-debug.log file.

What is the issue?

**James has to first manually create a empty terraform.log file.**

**Some issue with Terraform. Upgrade to latest version.**

**Your answer is correct**

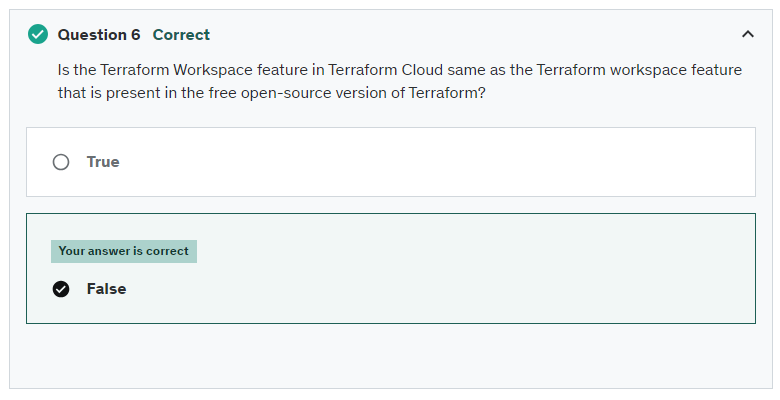
**Set the TF\_LOG environment variable**

**None of the Above**

**Overall explanation**

Even when TF\_LOG\_PATH is set, TF\_LOG must be set in order for any logging to be enabled.

<https://www.terraform.io/cli/config/environment-variables#tf_log_path>



**Question 7Correct**

There is a need to install nginx on the remote machine. Matthew has decided to make use of Terraform provisioner. The command to install nginx is as follows:

yum -y install nginx

Which among the below options is the right approach to achieve the use-case?

* provisioner "remote-exec" {
* inline = [
* command = "yum -y install nginx"
* ]
* }

**Your answer is correct**

* provisioner "remote-exec" {
* inline = [
* "yum -y install nginx"
* ]
* }
* provisioner "remote-exec" {
* inline = [
* run = "yum -y install nginx"
* ]
* }
* provisioner "local-exec" {
* inline = [
* "yum -y install nginx"
* ]
* }

**Overall explanation**

<https://www.terraform.io/docs/provisioners/remote-exec.html>

**Question 8Incorrect**

Which of the features are unique to Terraform Cloud Business Plan?

**Terraform Functions**

**Correct selection**

**Audit Logging**

**Sentinel**

**Your selection is correct**

**Clustering Functionality**

**Version Control Integration**

**Your selection is correct**

**Private Network Connectivity**

**Overall explanation**

[https://www.hashicorp.com/products/terraform/pricing/](https://www.terraform.io/docs/provisioners/remote-exec.html)

**Question 9Correct**

There is a requirement to rewrite Terraform configuration files to a canonical format and style.

Which Terraform command can be used?

**terraform plan**

**terraform validate**

**terraform format**

**terraform graph**

**Your answer is correct**

**terraform fmt**

**Overall explanation**

<https://www.terraform.io/cli/commands/fmt>

**Question 10Correct**

You want to contribute to the Terraform project. There are certain bugs that are reported in the Terraform binary, and you want to add a fix to it so that it is fixed in the newer Terraform version.

Which language will you need to write the fix?

**Java**

**Any Language**

**Python**

**Your answer is correct**

**Go**

**HCL**

**Overall explanation**

Terraform Core is a [statically-compiled binary](https://en.wikipedia.org/wiki/Static_build#Static_building) written in the [Go programming language](https://golang.org/). The compiled binary is the command line tool (CLI) terraform, the entrypoint for anyone using Terraform. The code is open source and hosted at github.com/hashicorp/terraform.

**Question 11Correct**

Matthew is using the Terraform Cloud Free version. Which of the below features is not supported in free version?

**Workspaces**

**Your answer is correct**

**Team Management**

**Private Module Registry**

**VCS Integration**

**Overall explanation**

<https://cloud.hashicorp.com/products/terraform/pricing>

**Question 12Correct**

Matthew has defined a map that contains multiple AMI ID associated with different regions. Following is a sample code:

* variable "ami\_ids" {
* type = "map"
* default = {
* "mumbai" = "image-abc"
* "germany" = "image-def"
* "states" = "image-xyz"
* }
* }

The user needs to select the image-def. Which one of the following methods be used to select that image?

**var.ami\_ids[1]**

**var.ami\_ids[2]**

**var.ami\_ids["image-def"]**

**Your answer is correct**

**var.ami\_ids["germany"]**

**Question 13Correct**

Which among the following is the default backend for the Terraform free version?

**consul**

**google cloud**

**S3**

**Your answer is correct**

**local**

**Overall explanation**

By default, Terraform uses a backend called [local](https://www.terraform.io/language/settings/backends/local), which stores state as a local file on disk. You can also configure one of the built-in backends.

**Question 14Correct**

Which Terraform files should be ignored by Git when committing code to a repository? (Choose Three)

**input.tf**

**output.tf**

**Your selection is correct**

**Files named exactly terraform.tfvars or terraform.tfvars.json.**

**Your selection is correct**

**terraform.tfstate**

**Your selection is correct**

**Any files with names ending in .auto.tfvars or .auto.tfvars.json.**

**Overall explanation**

.tfvars files, which are likely to contain sensitive data, such as passwords, private keys, and other secrets. These should not be part of version control as they are data points that are potentially sensitive and subject to change depending on the environment.

Along with this, even tfstate files should not be committed. It can contain sensitive data. Instead, it is recommended to have tfstate to a remote backend.

**Question 15Correct**

There are two workspaces which are created, namely PROD and QA

You want to deploy some infrastructure in the QA setup and have switched to QA workspace. Where will the state file be stored?

**.terraform**

**terraform.d**

**Your answer is correct**

**terraform.tfstate.d**

**terraform.tfstate**

**Overall explanation**

For local state, Terraform stores the workspace states in a directory called terraform.tfstate.d. This directory should be treated similarly to local-only terraform.tfstate; some teams commit these files to version control, although using a remote backend instead is recommended when there are multiple collaborators.

**Question 16Correct**

Terraform supports which of the following formats for it's configuration files (Choose 2)

**Python**

**Your selection is correct**

**HCL**

**Java**

**Your selection is correct**

**JSON**

**Go**

**C**

**Overall explanation**

<https://www.terraform.io/docs/language/syntax/json.html>

**Question 17Correct**

Following is the sample terraform code:

* output "db\_password" {
* value = aws\_db\_instance.db.password
* description = "RDS Password"
* sensitive = true
* }

Will the value associated with aws\_db\_instance.db.password be present within the terraform state file?

**Your answer is correct**

**True**

**False**

**Overall explanation**

Setting an output value in the root module as sensitive prevents Terraform from showing its value in the list of outputs at the end of terraform apply

Sensitive output values are still recorded in the state, and so will be visible to anyone who is able to access the state data.

**Question 18Correct**

Matthew intends to reference a VPC module from Git repository.

There is a requirement to use a specific branch instead of the default branch. What is the way to achieve this?

**Your answer is correct**

* module "vpc" {
* source = "git::https://kplabs.example.com/vpc.git?ref=v2.0.0"
* }
* module "vpc" {
* source = "git::https://kplabs.example.com/vpc.git?version=v2.0.0"
* }
* module "vpc" {
* source = "git::https://kplabsexample.com/vpc.git#reference=v2.0.0"
* }

**None of the Above**

**Overall explanation**

By default, Terraform will clone and use the default branch (referenced by HEAD) in the selected repository. You can override this using the ref argument.

<https://www.terraform.io/docs/language/modules/sources.html#generic-git-repository>

**Question 19Correct**

There is an EC2 instance that has been created manually in AWS.

There is a need to import that EC2 instance so that it can be managed by terraform going forward. Following is the configuration file written:

* resource "aws\_instance" "myec2" {
* ami = "ami-bf5540df"
* instance\_type = "t2.micro"
* vpc\_security\_group\_ids = ["sg-6ae7d613", "sg-53370035"]
* key\_name = "remotepractical"
* subnet\_id = "subnet-9e3cfbc5"
* tags {
* Name = "manual"
* }
* }

The instance-id associated with manually created EC2 is i-234567. How can the import process happen?

**terraform import i-234567**

**terraform import i-234567 aws\_instance.myec2**

**terraform import myec2 i-234567**

**Your answer is correct**

**terraform import aws\_instance.myec2 i-234567**

**Overall explanation**

<https://developer.hashicorp.com/terraform/language/state/import>

**Question 20Correct**

Matthew intends to make use of 2 separate AWS regions to deploy two EC2 instances. He has specified the following configuration in the providers.tf file.

* provider "aws" { region = "us-east-1" }
* provider "aws" { region = "us-west-2" }

Running terraform plan leads to an error. How can Matthew fix the issue?

**There cannot be two different AWS provider configuration in same file.**

**Your answer is correct**

**Make use of alias**

**Change the region from us to eu**

**None of the Above**

**Overall explanation**

[https://www.terraform.io/language/providers/configuration#alias-multiple-provider-configuration](https://www.terraform.io/language/providers/configuration#alias-multiple-provider-configurations)s

**Question 21Correct**

Enterprise Corp has a requirement to capture the highest verbosity of Logs.

Which of the following environment variables need to be set to achieve this use case?

**TF\_LOG = DEBUG**

**TF\_LOG = HIGHEST**

**Your answer is correct**

**TF\_LOG = TRACE**

**TF\_LOG = 1**

**Overall explanation**

**TF\_LOG=TRACE** This will enable all log levels, including the most verbose level, TRACE.

<https://www.terraform.io/internals/debugging>

**Question 22Correct**

The following Terraform configuration file contains resource definition for Database (aws\_db\_instance) and EC2 instance (aws\_instance) in AWS.

* resource "aws\_db\_instance" "kplabs-db" {
* allocated\_storage = 20
* storage\_type = "gp2"
* engine = "mysql"
* }
* resource "aws\_instance" "kplabs-ec2" {
* ami = "ami-090fa75af13c156b4"
* instance\_type = "m5.large"
* }

EC2 instance will be configured with an application and application requires connectivity to Database before it can start. Hence Database instance should be created first before EC2.

What is the way to achieve this?

**Manually create an RDS resource first and then create EC2.**

**Your answer is correct**

**Specify an explicit dependency using the depends\_on attribute.**

**Terraform will automatically figure out the dependency.**

**This is not possible in terraform.**

**Question 23Incorrect**

There is a requirement to rename a resource block in Terraform with the intention of retaining the existing object but tracking it under a new name

Which Terraform command can be used without having the resource to be destroyed and recreated?

**Your answer is incorrect**

**terraform mv**

**Correct answer**

**terraform state mv**

**terraform plan**

**terraform plan mv**

**Overall explanation**

By default, Terraform will understand moving or renaming a resource configuration as a request to delete the old object and create a new object at the new address, and so terraform state mv allows you to override that interpretation by pre-emptively attaching the existing object to the new address in Terraform.

<https://www.terraform.io/cli/commands/state/mv>

**Question 24Correct**

For the Remote Exec Provisioners, which among the following are the supported connection types? (Choose 2)

**Your selection is correct**

**WinRM**

**ftp**

**SMB**

**RDP**

**UDP**

**Your selection is correct**

**ssh**

**Overall explanation**

The remote-exec provisioner requires a connection and supports both ssh and winrm.

<https://www.terraform.io/language/resources/provisioners/remote-exec>

**Question 25Correct**

Matthew is making use of zipmap function and has following code:

* zipmap(["a", "b"], [1, 2])

What will be the output?

* {
* a
* b
* 1
* 2
* }
* {
* ab12
* }

**Your answer is correct**

* {
* "a" = 1,
* "b" = 2,
* }

**None of the Above**

**Overall explanation**

<https://www.terraform.io/docs/configuration/functions/zipmap.html>

**Question 26Correct**

Matthew has configured AWS provider within his terraform code.

Where will be the associated plugins for that provider be stored?

**In the /home directory of Matthew**

**In the /home/plugins directory of Matthew**

**Your answer is correct**

**.terraform/plugins folder under the same directory as that of terraform code.**

**None of the above.**

**Question 27Correct**

Matthew has created 3 new workspaces named dev, stage and prod.

When he runs terraform workspace list command, there are total of 4 workspaces that are displayed

* default
* dev
* stage
* prod

Can Matthew delete the default workspace as it is not required for his use-case?

**True**

**Your answer is correct**

**False**

**Overall explanation**

Terraform starts with a **single workspace named "default"**. This workspace is special both because it is the default and also because it cannot ever be deleted.

<https://www.terraform.io/language/state/workspaces#using-workspaces>

**Question 28Correct**

What are the things that the terraform import command requires to run? (Choose 2)

**Your selection is correct**

**Resource from ID**

**Provider**

**Path**

**Your selection is correct**

**Resource address**

**Overall explanation**

Import will find the existing resource from ID and import it into your Terraform state at the given ADDRESS.

Example Syntax:

terraform import aws\_instance.example i-abcd1234

<https://www.terraform.io/cli/commands/import>

**Question 29Incorrect**

Matthew has created a new VPC module and he wants to publish the module to the Terraform registry.

What are the requirements to publish the module to the Public Registry?

**Your selection is incorrect**

**Module must be secure and must not have any sensitive contents.**

**Your selection is correct**

**Module repositories must use this three-part name format, terraform-provider-name**

**Module must support all the versions of Terraform.**

**Your selection is correct**

**elease tag names must be a semantic version, which can optionally be prefixed with a v**

**Your selection is correct**

**The module must be on GitHub and must be a public repo.**

**Overall explanation**

<https://www.terraform.io/registry/modules/publish#requirements>

**Question 30Correct**

If a module declares a variable with a default value, is there a need to explicitly set a variable value?

**True**

**Your answer is correct**

**False**

**Overall explanation**

Since default is already set, if no explicit value is defined, the default value will be used.