

100% Money Back
Guarantee

Vendor:HashiCorp

Exam Code:TA-003-P

Exam Name:HashiCorp Certified: Terraform Associate

Version:Demo

QUESTION 1

When should you write Terraform configuration files for existing infrastructure that you want to start managing with Terraform?

- A. You can import infrastructure without corresponding Terraform code
- B. Terraform will generate the corresponding configuration files for you
- C. Before you run terraform Import
- D. After you run terraform import

Correct Answer: C

You need to write Terraform configuration files for the existing infrastructure that you want to import into Terraform, otherwise Terraform will not know how to manage it. The configuration files should match the type and name of the resources that you want to import.

QUESTION 2

Which Terraform collection type should you use to store key/value pairs?

- A. Set
- B. Map
- C. Tuple
- D. list

Correct Answer: B

The Terraform collection type that should be used to store key/value pairs is map. A map is a collection of values that are accessed by arbitrary labels, called keys. The keys and values can be of any type, but the keys must be unique within a map. For example, `var = { key1 = "value1", key2 = "value2" }` is a map with two key/value pairs. Maps are useful for grouping related values together, such as configuration options or metadata. References = [Collection Types], [Map Type Constraints]

QUESTION 3

When does Sentinel enforce policy logic during a Terraform Cloud run?

- A. Before the plan phase
- B. During the plan phase
- C. Before the apply phase
- D. After the apply phase

Correct Answer: C

Sentinel policies are checked after the plan stage of a Terraform run, but before it can be confirmed or the terraform apply is executed³. This allows you to enforce rules on your infrastructure before it is created or modified.

QUESTION 4

Where does the Terraform local backend store its state?

- A. In the terraform file
- B. In the /tmp directory
- C. In the terraform,tfstate file
- D. In the user\\s terraform,state file

Correct Answer: C

This is where the Terraform local backend stores its state, by default, unless you specify a different file name or location in your configuration. The local backend is the simplest backend type that stores the state file on your local disk.

QUESTION 5

A Terraform provider is NOT responsible for:

- A. Exposing resources and data sources based on an API
- B. Managing actions to take based on resources differences
- C. Understanding API interactions with some service
- D. Provisioning infrastructure in multiple

Correct Answer: D

This is not a responsibility of a Terraform provider, as it does not make sense grammatically or logically. A Terraform provider is responsible for exposing resources and data sources based on an API, managing actions to take based on resource differences, and understanding API interactions with some service.

QUESTION 6

Which of the following methods, used to provision resources into a public cloud, demonstrates the concept of infrastructure as code?

- A. curl commands manually run from a terminal
- B. A sequence of REST requests you pass to a public cloud API endpoint Most Voted
- C. A script that contains a series of public cloud CLI commands

D. A series of commands you enter into a public cloud console

Correct Answer: C

The concept of infrastructure as code (IaC) is to define and manage infrastructure using code, rather than manual processes or GUI tools. A script that contains a series of public cloud CLI commands is an example of IaC, because it uses code to provision resources into a public cloud. The other options are not examples of IaC, because they involve manual or interactive actions, such as running curl commands, sending REST requests, or entering commands into a console. References = [Introduction to Infrastructure as Code with Terraform] and [Infrastructure as Code]

QUESTION 7

_____backends support state locking.

- A. All
- B. No
- C. Some
- D. Only local

Correct Answer: C

Some backends support state locking, which prevents other users from modifying the state file while a Terraform operation is in progress. This prevents conflicts and data loss. Not all backends support this feature, and you can check the documentation for each backend type to see if it does.

QUESTION 8

What Terraform command always causes a state file to be updated with changes that might have been made outside of Terraform?

- A. Terraform plan -refresh-only
- B. Terraform show -json
- C. Terraform apply -lock-false
- D. Terraform plan target-state

Correct Answer: A

This is the command that always causes a state file to be updated with changes that might have been made outside of Terraform, as it will only refresh the state file with the current status of the real resources, without making any changes to them or creating a plan.

QUESTION 9

It is best practice to store secret data in the same version control repository as your Terraform configuration.

- A. True
- B. False

Correct Answer: B

It is not a best practice to store secret data in the same version control repository as your Terraform configuration, as it could expose your sensitive information to unauthorized parties or compromise your security. You should use environment variables, vaults, or other mechanisms to store and provide secret data to Terraform.

QUESTION 10

Which of the following statements about Terraform modules is not true?

- A. Modules can call other modules
- B. A module is a container for one or more resources
- C. Modules must be publicly accessible
- D. You can call the same module multiple times

Correct Answer: C

This is not true, as modules can be either public or private, depending on your needs and preferences. You can use the Terraform Registry to publish and consume public modules, or use Terraform Cloud or Terraform Enterprise to host and manage private modules.

QUESTION 11

Which option cannot be used to keep secrets out of Terraform configuration files?

- A. A Terraform provider
- B. Environment variables
- C. A -var flag
- D. secure string

Correct Answer: D

A secure string is not a valid option to keep secrets out of Terraform configuration files. A secure string is a feature of AWS Systems Manager Parameter Store that allows you to store sensitive data encrypted with a KMS key. However, Terraform does not support secure strings natively and requires a custom data source to retrieve them. The other options are valid ways to keep secrets out of Terraform configuration files. A Terraform provider can expose secrets as data sources that can be referenced in the configuration. Environment variables can be used to set values for input variables that contain secrets. A -var flag can be used to pass values for input variables that contain secrets from the command line or a file. References = [AWS Systems Manager Parameter Store], [Terraform AWS Provider Issue #55], [Terraform Providers], [Terraform Input Variables]

QUESTION 12

Which command must you first run before performing further Terraform operations in a working directory?

- A. terraform import
- B. terraform workspace
- C. terraform plan
- D. terraform init

Correct Answer: D

terraform init is the first command that should be run after writing a new Terraform configuration or cloning an existing one from version control. It initializes a working directory containing Terraform configuration files and downloads any required providers and modules. The other commands are used for different purposes, such as importing existing resources, switching between workspaces, generating execution plans, etc.