



VCE & PDF

Pass4itSure.com

<https://www.pass4itsure.com/terraform-associate-003.html>

2024 Latest pass4itsure TERRAFORM-ASSOCIATE-003 PDF and VCE dumps

Download

# TERRAFORM-ASSOCIATE-003<sup>Q&As</sup>

HashiCorp Certified: Terraform Associate (003)

## Pass HashiCorp TERRAFORM-ASSOCIATE-003 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass4itsure.com/terraform-associate-003.html>

100% Passing Guarantee  
100% Money Back Assurance

Following Questions and Answers are all new published by HashiCorp  
Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers



**QUESTION 1**

Select the command that doesn't cause Terraform to refresh its state.

- A. Terraform destroy
- B. Terraform apply
- C. Terraform plan
- D. Terraform state list

Correct Answer: D

This is the command that does not cause Terraform to refresh its state, as it only lists the resources that are currently managed by Terraform in the state file. The other commands will refresh the state file before performing their operations, unless you use the `-refresh=false` flag.

---

**QUESTION 2**

You add a new resource to an existing Terraform configuration, but do not update the version constraint in the configuration. The existing and new resources use the same provider. The working contains a `.terraform.lock`, `hc1` file. How will Terraform choose which version of the provider to use?

- A. Terraform will use the version recorded in your lock file
- B. Terraform will use the latest version of the provider for the new resource and the version recorded in the lock file to manage existing resources
- C. Terraform will check your state file to determine the provider version to use
- D. Terraform will use the latest version of the provider available at the time you provision your new resource

Correct Answer: A

This is how Terraform chooses which version of the provider to use, when you add a new resource to an existing Terraform configuration, but do not update the version constraint in the configuration. The lock file records the exact version of each provider that was installed in your working directory, and ensures that Terraform will always use the same provider versions until you run `terraform init -upgrade` to update them.

---

**QUESTION 3**

Variables declared within a module are accessible outside of the module.

- A. True
- B. False

Correct Answer: B

Variables declared within a module are only accessible within that module, unless they are explicitly exposed as output

---



values1.

---

#### QUESTION 4

You should run terraform fmt to rewrite all Terraform configurations within the current working directory to conform to Terraform-style conventions.

- A. True
- B. False

Correct Answer: A

You should run terraform fmt to rewrite all Terraform configurations within the current working directory to conform to Terraform-style conventions. This command applies a subset of the Terraform language style conventions, along with other minor adjustments for readability. It is recommended to use this command to ensure consistency of style across different Terraform codebases. The command is optional, opinionated, and has no customization options, but it can help you and your team understand the code more quickly and easily. References = : Command: fmt : Using Terraform fmt Command to Format Your Terraform Code

---

#### QUESTION 5

What does Terraform not reference when running a terraform apply -refresh-only ?

- A. State file
- B. Credentials
- C. Cloud provider
- D. Terraform resource definitions in configuration files

Correct Answer: D

When running a terraform apply -refresh-only, Terraform does not reference the configuration files, but only the state file, credentials, and cloud provider. The purpose of this command is to update the state file with the current status of the real resources, without making any changes to them1.

---

#### QUESTION 6

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 7

What information does the public Terraform Module Registry automatically expose about published modules?

- A. Required input variables
- B. Optional inputs variables and default values
- C. Outputs
- D. All of the above
- E. None of the above

Correct Answer: D

The public Terraform Module Registry automatically exposes all the information about published modules, including required input variables, optional input variables and default values, and outputs. This helps users to understand how to use and configure the modules.

---

### QUESTION 8

\_\_\_\_\_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 9

Terraform providers are part of the Terraform core binary.

- A. True
- B. False

Correct Answer: B



Terraform providers are not part of the Terraform core binary. Providers are distributed separately from Terraform itself and have their own release cadence and version numbers. Providers are plugins that Terraform uses to interact with various APIs, such as cloud providers, SaaS providers, and other services. You can find and install providers from the Terraform Registry, which hosts providers for most major infrastructure platforms. You can also load providers from a local mirror or cache, or develop your own custom providers. To use a provider in your Terraform configuration, you need to declare it in the provider requirements block and optionally configure its settings in the provider block.

References = : Providers - Configuration Language | Terraform : Terraform Registry

-Providers Overview | Terraform

---

### QUESTION 10

You're building a CI/CD (continuous integration/continuous delivery) pipeline and need to inject sensitive variables into your Terraform run. How can you do this safely?

- A. Copy the sensitive variables into your Terraform code
- B. Store the sensitive variables in a `secure_vars.tf` file
- C. Store the sensitive variables as plain text in a source code repository
- D. Pass variables to Terraform with a `-var` flag

Correct Answer: D

This is a secure way to inject sensitive variables into your Terraform run, as they will not be stored in any file or source code repository. You can also use environment variables or variable files with encryption to pass sensitive variables to Terraform.

---

### QUESTION 11

When do changes invoked by terraform apply take effect?

- A. After Terraform has updated the state file
- B. Once the resource provider has fulfilled the request
- C. Immediately
- D. None of the above are correct

Correct Answer: B

Changes invoked by terraform apply take effect once the resource provider has fulfilled the request, not after Terraform has updated the state file or immediately. The state file is only a reflection of the real resources, not a source of truth.

---

### QUESTION 12

What does the default "local" Terraform backend store?

- A. `tfplan` files



- B. State file
- C. Provider plugins
- D. Terraform binary

Correct Answer: B

The default "local" Terraform backend stores the state file in a local file named terraform.tfstate, which can be used to track and manage the state of your infrastructure3.

---

### QUESTION 13

You have declared a variable called var.list which is a list of objects that all have an attribute id . Which options will produce a list of the IDs? Choose two correct answers.

- A. [ var.list [ \* ] , id ]
- B. [ for o in var.list : o.id ]
- C. var.list[\*].id
- D. { for o in var.llst : o => o.id }

Correct Answer: BC

These are two ways to produce a list of the IDs from a list of objects that have an attribute id, using either a for expression or a splat expression syntax.

---

### QUESTION 14

You can reference a resource created with for\_each using a Splat ( \* ) expression.

- A. True
- B. False

Correct Answer: B

You cannot reference a resource created with for\_each using a splat ( \* ) expression, as it will not work with resources that have non-numeric keys. You need to use a for expression instead to iterate over the resource instances.

---

### QUESTION 15

In a Terraform Cloud workspace linked to a version control repository speculative plan run start automatically commit changes to version control.

- A. True
- B. False



Correct Answer: A

When you use a remote backend that needs authentication, HashiCorp recommends that you:

[TERRAFORM-ASSOCIATE-003 Practice Test](#)

[TERRAFORM-ASSOCIATE-003 Study Guide](#)

[TERRAFORM-ASSOCIATE-003 Braindumps](#)