

# DUMPSQUEEN

## HashiCorp Certified: Terraform Associate

HashiCorp TA-002-P

Version Demo

Total Demo Questions: 20

Total Premium Questions: 449

Buy Premium PDF

<https://dumpsqueen.com>

[support@dumpsqueen.com](mailto:support@dumpsqueen.com)

dumpsqueen.com

## Topic Break Down

Topic	No. of Questions
Topic 1, Exam Set A	100
Topic 2, Exam Set B	57
Topic 3, Exam Set C	72
Topic 4, Exam Set D	220
Total	449

## QUESTION NO: 1

When using Terraform to deploy resources into Azure, which scenarios are true regarding state files? (Choose two.)

- A. When a change is made to the resources via the Azure Cloud Console, the changes are recorded in a new state file
- B. When a change is made to the resources via the Azure Cloud Console, Terraform will update the state file to reflect them during the next plan or apply
- C. When a change is made to the resources via the Azure Cloud Console, the current state file will not be updated
- D. When a change is made to the resources via the Azure Cloud Console, the changes are recorded in the current state file

**ANSWER: B C**

## QUESTION NO: 2

Which one is the right way to import a local module names consul?

- A. `module "consul" { source = "consul" }`
- B. `module "consul" { source = "./consul" }`
- C. `module "consul" { source = "../consul" }`
- D. `module "consul" { source = "module/consul" }`

**ANSWER: B C**

### Explanation:

A local path must begin with either `./` or `../` to indicate that a local path is intended, to distinguish from a module registry address.

```
module "consul" {  
  source = "./consul"  
}
```

## QUESTION NO: 3

What does terraform import allow you to do?

- A. Import a new Terraform module
- B. Use a state file to import infrastructure to the cloud

- C. Import provisioned infrastructure to your state file
- D. Import an existing state file to a new Terraform workspace

**ANSWER: C**

## QUESTION NO: 4

You have created a custom variable definition file testing.tfvars. How will you use it for provisioning infrastructure?

- A. terraform apply -var-state-file ="testing.tfvars"
- B. terraform plan -var-file="testing.tfvar"
- C. terraform apply -var-file="testing.tfvars"
- D. terraform apply var-file="testing.tfvars"

**ANSWER: C**

**Explanation:**

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

## QUESTION NO: 5

Workspaces in Terraform provides similar functionality in the open-source, Terraform Cloud, and Enterprise versions of Terraform.

- A. True
- B. False

**ANSWER: B**

**Explanation:**

<https://www.terraform.io/docs/cloud/migrate/workspaces.html>

Workspaces, managed with the terraform workspace command, aren't the same thing as Terraform Cloud's workspaces. Terraform Cloud workspaces act more like completely separate working directories; CLI workspaces are just alternate state files.

## QUESTION NO: 6

Which of the below terraform commands do not run terraform refresh implicitly before taking actual action of the command?

- A. terraform apply

- B. terraform destroy
- C. terraform init
- D. terraform import
- E. terraform plan

**ANSWER: C D**

**Explanation:**

<https://www.terraform.io/docs/commands/refresh.html>

## QUESTION NO: 7

Which of the following command can be used to view the specified version constraints for all providers used in the current configuration.

- A. terraform providers
- B. terraform state show
- C. terraform provider
- D. terraform plan

**ANSWER: A**

**Explanation:**

Use the terraform providers command to view the specified version constraints for all providers used in the current configuration.

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

## QUESTION NO: 8

Which of the following Terraform commands will automatically refresh the state unless supplied with additional flags or arguments? Choose TWO correct answers.

- A. terraform state
- B. terraform apply
- C. terraform plan
- D. terraform validate
- E. terraform output

**ANSWER: B C**

## QUESTION NO: 9

Select all Operating Systems that Terraform is available for. (select five)

- A. Linux
- B. macOS
- C. Unix
- D. Solaris
- E. Windows
- F. FreeBSD

**ANSWER: A B D E F**

### Explanation:

Explanation

Terraform is available for macOS, FreeBSD, OpenBSD, Linux, Solaris, Windows <https://www.terraform.io/downloads.html>

## QUESTION NO: 10

Select the feature below that best completes the sentence:

The following list represents the different types of \_\_\_\_\_ available in Terraform.

1. max
  2. min
  3. join
  4. replace
  5. list
  6. length
  7. range
- A. Backends
  - B. Data sources
  - C. Named values
  - D. Functions

**ANSWER: D**

**Explanation:**

The Terraform language includes a number of built-in functions that you can call from within expressions to transform and combine values. The Terraform language does not support user-defined functions, and only the functions built into the language are available for use.

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

## QUESTION NO: 11

From the answers below, select the advantages of using Infrastructure as Code.

- A. Provide a codified workflow to develop customer-facing applications.
- B. Safely test modifications using a "dry run" before applying any actual changes.
- C. Easily integrate with application workflows (GitLab Actions, Azure DevOps, CI/CD tools).
- D. Easily change and update existing infrastructure.
- E. Provide reusable modules for easy sharing and collaboration.

**ANSWER: B C D E**

**Explanation:**

Infrastructure as Code is not used to develop applications, but it can be used to help deploy or provision those applications to a public cloud provider or on-premises infrastructure.

All of the others are benefits to using Infrastructure as Code over the traditional way of managing infrastructure, regardless if it's public cloud or on-premises.

## QUESTION NO: 12

What is terraform refresh intended to detect?

- A. Terraform configuration code changes
- B. Empty state files
- C. State file drift
- D. Corrupt state files

**ANSWER: C**

**Explanation:**

"The terraform refresh command reads the current settings from all managed remote objects and updates the Terraform state to match. Warning: This command is deprecated, because its default behavior is unsafe if you have misconfigured

credentials for any of your providers. See below for more information and recommended alternatives."  
<https://www.terraform.io/cli/commands/refresh>

## QUESTION NO: 13

In contrast to Terraform Open Source, when working with Terraform Enterprise and Cloud Workspaces, conceptually you could think about them as completely separate working directories.

- A. True
- B. False

**ANSWER: A**

**Explanation:**

<https://www.terraform.io/cloud-docs/workspaces>

"When run locally, Terraform manages each collection of infrastructure with a persistent working directory, which contains a configuration, state data, and variables. Since Terraform CLI uses content from the directory it runs in, you can organize infrastructure resources into meaningful groups by keeping their configurations in separate directories.

## QUESTION NO: 14

Which of the following challenges would Terraform be a candidate for solving? (Select THREE)

- A. Enable self-service infrastructure to allocate resources on your proprietary private cloud.
- B. Reduce the number of workflows needed for managing infrastructure across each of the companies public and private clouds.
- C. Utilize a single tool for all of the infrastructure and configuration management needs.
- D. Have a single interoperable tool to manage the variety of services including GitHub repositories, MySQL database, and Kubernetes clusters.

**ANSWER: A B D**

## QUESTION NO: 15

Which two steps are required to provision new infrastructure in the Terraform workflow? (Choose two.)

- A. Destroy
- B. Apply
- C. Import
- D. Init



E. Validate

**ANSWER: B D**

**Explanation:**

Reference: <https://www.terraform.io/guides/core-workflow.html>

## QUESTION NO: 16

Refer below code where pessimistic constraint operator has been used to specify a version of a provider.

```
terraform { required_providers { aws = "~> 1.1.0" }}
```

Which of the following options are valid provider versions that satisfy the above constraint. (select two)

- A. 1.1.1
- B. 1.2.9
- C. 1.1.8
- D. 1.2.0

**ANSWER: A C**

**Explanation:**

Pessimistic constraint operator, constraining both the oldest and newest version allowed. For example, `~> 0.9` is equivalent to `>= 0.9, < 1.0`, and `~> 0.8.4`, is equivalent to `>= 0.8.4, < 0.9`

## QUESTION NO: 17

What command can you run to generate DOT (Document Template) formatted data to visualize Terraform dependencies?

- A. terraform refresh
- B. terraform show
- C. terraform graph
- D. terraform output

**ANSWER: C**

**Explanation:**

The terraform graph command is used to generate a visual representation of either a configuration or execution plan. The output is in the DOT format, which can be used by GraphViz to generate charts.

## QUESTION NO: 18

Which of the following connection types are supported by the remote-exec provisioner? (select two)

- A. WinRM
- B. UDP
- C. SMB
- D. RDP
- E. ssh

**ANSWER: A E**

### Explanation:

The remote-exec provisioner invokes a script on a remote resource after it is created. The remote-exec provisioner supports both ssh and winrm type connections.

remote-exec connection types -

\* ssh on Linux

\* winrm on Windows

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

## QUESTION NO: 19

State locking does not happen automatically and must be specified at run

- A. False
- B. True

**ANSWER: A**

### Explanation:

State locking happens automatically on all operations that could write state.

<https://www.terraform.io/docs/state/locking.html>

## QUESTION NO: 20

Do terraform workspaces help in adding/allowing multiple state files for a single configuration?

- A. True
- B. False

**ANSWER: A**