



Vendor: HashiCorp

Exam Code: TA-002-P

Exam Name: HashiCorp Certified: Terraform Associate

Version: DEMO

QUESTION 1

How is the Terraform remote backend different than other state backends such as S3, Consul, etc.?

- A. It can execute Terraform runs on dedicated infrastructure on premises or in Terraform Cloud
- B. It doesn't show the output of a terraform apply locally
- C. It is only available to paying customers
- D. All of the above

Answer: A

Explanation:

If you and your team are using Terraform to manage meaningful infrastructure, we recommend using the remote backend with Terraform Cloud or Terraform Enterprise.

Reference: <https://www.terraform.io/docs/language/settings/backends/index.html>

QUESTION 2

Which provisioner invokes a process on the resource created by Terraform?

- A. remote-exec
- B. null-exec
- C. local-exec
- D. file

Answer: A

Explanation:

The remote-exec provisioner invokes a script on a remote resource after it is created.

Reference:

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

QUESTION 3

What command does Terraform require the first time you run it within a configuration directory?

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

Answer: B

Explanation:

terraform init command is used to initialize a working directory containing Terraform configuration files.

Reference: <https://www.terraform.io/docs/cli/commands/init.html>

QUESTION 4

You have provisioned some virtual machines (VMs) on Google Cloud Platform (GCP) using the gcloud command line tool. However, you are standardizing with Terraform and want to manage these VMs using Terraform instead.

What are the two things you must do to achieve this? (Choose two.)

- A. Provision new VMs using Terraform with the same VM names
- B. Use the terraform import command for the existing VMs
- C. Write Terraform configuration for the existing VMs
- D. Run the terraform import-gcp command

Answer: BD

Explanation:

The terraform import command is used to import existing infrastructure. Import existing Google Cloud resources into Terraform with Terraformer.

<https://www.terraform.io/docs/cli/import/usage.html>

<https://cloud.google.com/docs/terraform>

QUESTION 5

Why would you use the terraform taint command?

- A. When you want to force Terraform to destroy a resource on the next apply
- B. When you want to force Terraform to destroy and recreate a resource on the next apply
- C. When you want Terraform to ignore a resource on the next apply
- D. When you want Terraform to destroy all the infrastructure in your workspace

Answer: B

Explanation:

The terraform taint command manually marks a Terraform-managed resource as tainted, forcing it to be destroyed and recreated on the next apply.

Reference: <https://www.terraform.io/docs/cli/commands/taint.html>

QUESTION 6

When should you use the force-unlock command?

- A. You see a status message that you cannot acquire the lock
- B. You have a high priority change
- C. Automatic unlocking failed
- D. Your apply failed due to a state lock

Answer: C

Explanation:

Manually unlock the state for the defined configuration.

<https://www.terraform.io/docs/cli/commands/force-unlock.html>

QUESTION 7

terraform validate validates the syntax of Terraform files.

- A. True
- B. False

Answer: A

Explanation:

The terraform validate command validates the syntax and arguments of the Terraform configuration files.

Reference: <https://www.terraform.io/docs/cli/code/index.html>

QUESTION 8

A Terraform provisioner must be nested inside a resource configuration block.

- A. True
- B. False

Answer: A

Explanation:

Most provisioners require access to the remote resource via SSH or WinRM, and expect a nested connection block with details about how to connect.

<https://www.terraform.io/docs/language/resources/provisioners/connection.html>

QUESTION 9

What does the default "local" Terraform backend store?

- A. tfplan files
- B. Terraform binary
- C. Provider plugins
- D. State file

Answer: D

Explanation:

The local backend stores state on the local filesystem, locks that state using system APIs, and performs operations locally.

<https://www.terraform.io/docs/language/settings/backends/local.html>

QUESTION 10

You have multiple team members collaborating on infrastructure as code (IaC) using Terraform, and want to apply formatting standards for readability.

How can you format Terraform HCL (HashiCorp Configuration Language) code according to standard Terraform style convention?

- A. Run the terraform fmt command during the code linting phase of your CI/CD process
- B. Designate one person in each team to review and format everyone's code
- C. Manually apply two spaces indentation and align equal sign "=" characters in every Terraform file (*.tf)
- D. Write a shell script to transform Terraform files using tools such as AWK, Python, and sed

Answer: C

Explanation:

Indent two spaces for each nesting level.

When multiple arguments with single-line values appear on consecutive lines at the same nesting level, align their equals signs.

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

QUESTION 11

What value does the Terraform Cloud/Terraform Enterprise private module registry provide over the public Terraform Module Registry?

- A. The ability to share modules with public Terraform users and members of Terraform Enterprise Organizations
- B. The ability to tag modules by version or release
- C. The ability to restrict modules to members of Terraform Cloud or Enterprise organizations
- D. The ability to share modules publicly with any user of Terraform

Answer: D

Explanation:

Terraform Registry is an index of modules shared publicly using this protocol. This public registry is the easiest way to get started with Terraform and find modules created by others in the community.

<https://www.terraform.io/docs/language/modules/sources.html>

QUESTION 12

Which argument(s) is (are) required when declaring a Terraform variable?

- A. type
- B. default
- C. description
- D. All of the above
- E. None of the above

Answer: B

Explanation:

The variable declaration can also include a default argument.

Reference: <https://www.terraform.io/docs/language/values/variables.html>

QUESTION 13

Where does the Terraform local backend store its state?

- A. In the /tmp directory
- B. In the terraform.tfvars file
- C. In the terraform.tfstate file
- D. In the user's .terraformrc file

Answer: C

Explanation:

The local backend stores state on the local filesystem, locks that state using system APIs, and performs operations locally.

Reference: <https://www.terraform.io/docs/language/settings/backends/local.html>

QUESTION 14

Only the user that generated a plan may apply it.

- A. True
- B. False

Answer: A

Explanation:

The optional -out argument can be used to save the generated plan to a file for later execution with terraform apply, which can be useful when running Terraform in automation.
Reference: <https://learn.hashicorp.com/tutorials/terraform/automate-terraform>

QUESTION 15

Examine the following Terraform configuration, which uses the data source for an AWS AMI.

What value should you enter for the ami argument in the AWS instance resource?

```
data "aws_ami" "ubuntu" {  
  ...  
}  
  
resource "aws_instance" "web" {  
  ami = _____  
  instance_type = "t2.micro"  
  
  tags = {  
    Name = "HelloWorld"  
  }  
}
```

- A. aws_ami.ubuntu
- B. data.aws_ami.ubuntu
- C. data.aws_ami.ubuntu.id
- D. aws_ami.ubuntu.id

Answer: C

Explanation:

```
resource "aws_instance" "web" {  
  ami = data.aws_ami.ubuntu.id  
  https://registry.terraform.io/providers/hashicorp/aws/latest/docs/resources/instance
```

QUESTION 16

Where in your Terraform configuration do you specify a state backend?

- A. The terraform block
- B. The resource block
- C. The provider block
- D. The datasource block

Answer: A

Explanation:

Backends are configured with a nested backend block within the top-level terraform block.
<https://www.terraform.io/docs/language/settings/backends/configuration.html>

QUESTION 17

Terraform providers are always installed from the Internet.

- A. True
- B. False

Answer: B

Explanation:

Terraform configurations must declare which providers they require, so that Terraform can install and use them.

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

QUESTION 18

Which of these is the best practice to protect sensitive values in state files?

- A. Blockchain
- B. Secure Sockets Layer (SSL)
- C. Enhanced remote backends
- D. Signed Terraform providers

Answer: C

Explanation:

Use of remote backends and especially the availability of Terraform Cloud, there are now a variety of backends that will encrypt state at rest and will not store the state in cleartext on machines running.

Reference: <https://www.terraform.io/docs/extend/best-practices/sensitive-state.html>

QUESTION 19

terraform refresh command will not modify infrastructure, but does modify the state file.

- A. True
- B. False

Answer: A

Explanation:

The terraform refresh command is used to reconcile the state Terraform knows about (via its state file) with the real-world infrastructure. This can be used to detect any drift from the last-known state, and to update the state file. This does not modify infrastructure, but does modify the state file.

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

QUESTION 20

Which of the following clouds does not have a provider maintained HashiCorp?

- A. IBM Cloud
- B. DigitalOcean
- C. OpenStack
- D. AWS

Answer: A

Explanation:

IBM Cloud does not have a provider maintained by HashiCorp, although IBM Cloud does maintain their own Terraform provider.

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

QUESTION 21

Environment variables can be used to set variables. The environment variables must be in the format "____"____<variablename>. Select the correct prefix string from the following list.

- A. TF_CLI_ARGS
- B. TF_VAR
- C. TF_VAR_
- D. TF_VAR_ENV

Answer: C

Explanation:

Environment variables can be used to set variables. The environment variables must be in the format TF_VAR_name and this will be checked last for a value.

For example:

```
export TF_VAR_region=us-west-1
```

```
export TF_VAR_ami=ami-049d8641
```

```
export TF_VAR_alist='[1,2,3]'
```

```
export TF_VAR_amap='{ foo = "bar", baz = "qux" }'
```

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

QUESTION 22

Terraform must track metadata such as resource dependencies. Where is this data stored?

- A. workspace
- B. backend
- C. state file
- D. metadata store

Answer: C

Explanation:

Terraform typically uses the configuration to determine dependency order. However, when you delete a resource from a Terraform configuration, Terraform must know how to delete that resource. Terraform can see that a mapping exists for a resource not in your configuration and plan to destroy. However, since the configuration no longer exists, the order cannot be determined from the configuration alone.

To ensure correct operation, Terraform retains a copy of the most recent set of dependencies within the state. Now Terraform can still determine the correct order for destruction from the state when you delete one or more items from the configuration.

<https://www.terraform.io/docs/state/purpose.html#metadata>

Thank You for Trying Our Product

Lead2pass Certification Exam Features:

- ★ More than **99,900** Satisfied Customers Worldwide.
- ★ Average **99.9%** Success Rate.
- ★ **Free Update** to match latest and real exam scenarios.
- ★ **Instant Download** Access! No Setup required.
- ★ Questions & Answers are downloadable in **PDF** format and **VCE** test engine format.
- ★ Multi-Platform capabilities - **Windows, Laptop, Mac, Android, iPhone, iPod, iPad**.
- ★ **100%** Guaranteed Success or **100%** Money Back Guarantee.
- ★ **Fast**, helpful support **24x7**.



View list of all certification exams: <http://www.lead2pass.com/all-products.html>



Microsoft



ORACLE



CITRIX



JUNIPER
NETWORKS



EMC²
where information lives

10% Discount Coupon Code: ASTR14