

Terraform-Associate-004 Exam Answers & Terraform-Associate-004 Valid Practice Questions

TERRAFORM ASSOCIATE EXAM COMPLETE WITH CORRECT ANSWERS

The terraform.tfstate file always matches your currently built infrastructure.

- A. True
- B. False

B

One remote backend configuration always maps to a single remote workspace.

- A. True
- B. False

B

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

A

What is the workflow for deploying new infrastructure with Terraform?

- A. terraform plan to import the current infrastructure to the state file, make code changes, and terraform apply to update the infrastructure.
- B. Write a Terraform configuration, run terraform show to view proposed changes, and terraform apply to create new infrastructure.
- C. terraform import to import the current infrastructure to the state file, make code changes, and terraform apply to update the infrastructure.
- D. Write a Terraform configuration, run terraform init, run terraform plan to view planned infrastructure changes, and terraform apply to create new infrastructure.

D

A provider configuration block is required in every Terraform configuration. Example:

DOWNLOAD the newest Pass4training Terraform-Associate-004 PDF dumps from Cloud Storage for free:
https://drive.google.com/open?id=1Uha7B1Hgb23JjXB0MRDbDmHTy0h_rpY8

In order to meet different needs of the candidates, three versions for Terraform-Associate-004 exam materials are available. You can choose the one you prefer for your training. Terraform-Associate-004 PDF version is printable, and you can print them into hard one if you like. Terraform-Associate-004 Soft test engine can install in more than 200 personal computers, it also support MS operating system. Terraform-Associate-004 Online Test engine can is convenient and easy to learn, it supports all web browsers, and you can have a general review of what you have learned through this version.

For candidates, one of the most important things for you is to know the latest information of the exam. Terraform-Associate-004 Training Materials of us will meet your needs. And our system will send the latest version to you automatically, so that you can know the recent information. We have free update for one year, that is to say, you can get free update version for 365 days after purchasing. In addition, we will pass guarantee and money back guarantee.

>> Terraform-Associate-004 Exam Answers <<

Professional Terraform-Associate-004 Exam Answers - Fantastic Terraform-Associate-004 Exam Tool Guarantee Purchasing Safety

Our Terraform-Associate-004 learning guide is for the world and users are very extensive. In order to give users a better experience, we have been constantly improving. The high quality and efficiency of Terraform-Associate-004 exam prep has been recognized by users. The high passing rate of our Terraform-Associate-004 test materials are its biggest feature. As long as you use

Terraform-Associate-004 Exam Prep, you can certainly harvest what you want thing. Not only you can pass the Terraform-Associate-004 exam in the shortest time, but also you can obtain the dreaming Terraform-Associate-004 certification to have a brighter future.

HashiCorp Terraform-Associate-004 Exam Syllabus Topics:

| Topic | Details |
|---------|---|
| Topic 1 | <ul style="list-style-type: none">• Terraform state management: This domain focuses on managing Terraform's state file, understanding local and remote backends, implementing state locking, and handling resource drift. |
| Topic 2 | <ul style="list-style-type: none">• Terraform configuration: This domain covers writing Terraform code including resources and data blocks, using variables and outputs, handling complex types, creating dynamic configurations with expressions and functions, managing dependencies, implementing validation, and handling sensitive data. |
| Topic 3 | <ul style="list-style-type: none">• Infrastructure as Code (IaC) with Terraform: This domain covers the foundational concept of Infrastructure as Code and how Terraform enables managing resources across multiple cloud providers and services through a unified workflow. |
| Topic 4 | <ul style="list-style-type: none">• Maintain infrastructure with Terraform: This domain addresses importing existing infrastructure into Terraform, inspecting state using CLI commands, and using verbose logging for troubleshooting. |

HashiCorp Certified: Terraform Associate (004) (HCTA0-004) Sample Questions (Q41-Q46):

NEW QUESTION # 41

Which of the following commands would you use to access all of the attributes and details of a resource managed by Terraform?

- A. terraform state list 'provider_type.name'
- B. terraform state list
- C. terraform state show 'provider_type.name'
- D. terraform get 'provider_type.name'

Answer: C

Explanation:

The terraform state show command allows you to access all of the attributes and details of a resource managed by Terraform. You can use the resource address (e.g. provider_type.name) as an argument to show the information about a specific resource. The terraform state list command only shows the list of resources in the state, not their attributes. The terraform get command downloads and installs modules needed for the configuration. It does not show any information about resources. References = [Command: state show] and [Command: state list]

NEW QUESTION # 42

The Terraform binary version and provider versions must match each other in a single configuration.

- A. True
- B. False

Answer: B

Explanation:

The Terraform binary version and provider versions do not have to match each other in a single configuration. Terraform allows you to specify provider version constraints in the configuration's terraform block, which can be different from the Terraform binary version¹. Terraform will use the newest version of the provider that meets the configuration's version constraints². You can also use the dependency lock file to ensure Terraform is using the correct provider version³. Reference =

* 1: Providers - Configuration Language | Terraform | HashiCorp Developer

* 2: Multiple provider versions with Terraform - Stack Overflow

NEW QUESTION # 43

What is the purpose of the terraform.lock.hcl file in Terraform?

- A. Tracking specific provider dependencies.
- B. There is no such file.
- C. Storing references to workspaces, which are locked.
- D. Preventing Terraform runs from occurring.

Answer: A

Explanation:

Rationale for Correct Answer (D):

The terraform.lock.hcl file is automatically created and maintained by Terraform. It records the specific versions and checksums of providers used in a configuration, ensuring consistent runs across environments and machines.

Analysis of Incorrect Options:

A . There is no such file: Incorrect - the lock file exists and is crucial for dependency management.

B . Storing references to workspaces: Incorrect - workspaces are tracked in the state file, not the lock file.

C . Preventing Terraform runs: Incorrect - the lock file does not block runs; it enforces consistent provider versions.

Key Concept:

The terraform.lock.hcl file enforces provider version consistency, which is critical for reproducible and reliable Terraform deployments.

Reference:

Terraform Exam Objective - Read, Generate, and Modify Configurations.

NEW QUESTION # 44

Which of the following is not a valid Terraform variable type?

- A. map
- B. list
- C. array
- D. string

Answer: C

Explanation:

This is not a valid Terraform variable type. The other options are valid variable types that can store different kinds of values.

NEW QUESTION # 45

Infrastructure as Code (IaC) can be stored in a version control system along with application code.

- A. False
- B. True

Answer: B

Explanation:

Infrastructure as Code (IaC) can indeed be stored in a version control system along with application code.

This practice is a fundamental principle of modern infrastructure management, allowing teams to apply software development practices like versioning, peer review, and CI/CD to infrastructure management.

Storing IaC configurations in version control facilitates collaboration, history tracking, and change management. References = While this concept is a foundational aspect of IaC and is widely accepted in the industry, direct references from the HashiCorp Terraform Associate (003) study materials were not found in the provided files. However, this practice is encouraged in Terraform's best practices and various HashiCorp learning resources.

