

HashiCorp.Terraform-Associate-003.v2024-04-29.q107

□□□□:	Terraform-Associate-003
□□□□:	HashiCorp Certified: Terraform Associate (003) (HCTA0-003)
□□□:	HashiCorp
□□ □□ □□□:	107
□□:	v2024-04-29
# □□ □:	431
# □□ □□□:	1070
https://www.krdump.com/HashiCorp.Terraform-Associate-003.v2024-04-29.q107.html	

NEW QUESTION: 1

Terraform □□ □□□ □□ □□□□ Amazon S3□ □□□□□ □□□□□□. backend.tf□□ □ □□ □□ □□□ □□□□□.

```
terraform {  
  backend "s3" {  
    bucket = "my-tf-bucket"  
    region = "us-east-1"  
  }  
}
```



□□ □□ □□□ □□□ S3 □□ □□□□ □□□□□□□□ □□□ □□□□□?

- A. □□□□ □□
- B. □□□ □□□
- C. Terraform □□
- D. Terraform □□ □□

Answer: B ([LEAVE A REPLY](#))

□ □□□ □ □□□□ □□□□□ □□ □□ □□□ □ □□□ □□□□□□□□□□ □□□□ □ □□□□3. □□ □□□ □ □□□ □□□ □□□□.

NEW QUESTION: 2

□□ □ Terraform□ □□□ □□□ □□□ □□ □□ □□□□□?

- A. □□
- B. □□
- C. □□
- D. □□□□

Answer: A ([LEAVE A REPLY](#))

□□

TF_LOG = [Terraform TRACE, INFO, WARN, ERROR, DEBUG, INFO, TRACE, INFO, WARN, ERROR, DEBUG] stdout = [Terraform TRACE, INFO, WARN, ERROR, DEBUG]

NEW QUESTION: 9

TF_LOG = [Terraform TRACE, INFO, WARN, ERROR, DEBUG] stdout = [Terraform TRACE, INFO, WARN, ERROR, DEBUG]

- A. TRACE
- B. INFO

Answer: A (LEAVE A REPLY)

TF_LOG = [Terraform TRACE, INFO, WARN, ERROR, DEBUG] stdout = [Terraform TRACE, INFO, WARN, ERROR, DEBUG]

NEW QUESTION: 10

Terraform import command is used to import existing resources into Terraform state.

- A. terraform import
- B. terraform import -state
- C. terraform import -state-dir
- D. terraform import -state-dir

Answer: C (LEAVE A REPLY)

Terraform import command is used to import existing resources into Terraform state.

NEW QUESTION: 11

Terraform import command is used to import existing resources into Terraform state.

- A. terraform import
- B. terraform import -state

Answer: B (LEAVE A REPLY)

Terraform import command is used to import existing resources into Terraform state.

NEW QUESTION: 12

TF_LOG = [Terraform TRACE, INFO, WARN, ERROR, DEBUG] stdout = [Terraform TRACE, INFO, WARN, ERROR, DEBUG]

- A. Terraform rm:aws_instance.ubuntu[1]
- B. Terraform rm:aws_instance.ubuntu[1]

□ □□□□ □□ □ □□□ □□ □□□□ □□□ □ □□□□. □□ □□ □□□
□□ □□ □□□ □□□ □□□ □□□ □□ □□□□ □□□ □□□□ □□□. □□□
□ □□□□ □□ force-unlock □□□□ □□□ □□ ID□ □□□□□. □□ □□□ □□□□
Terraform□ □ □□ ID□ □□□□□. □ □□ ID□ nonce □□□ □□ □□ □ □□ □□□ □□
□ □□□ □□□□ □□□ □□□□□. □□ □□□ force-unlock □□□ □□□□□ □□□ □□□
□□□□. □□□□ □□□ □□ □□, □□ □□□□ □□ □□□ □□□ □□ □□ □□□ □□□
□ □□□ □□ □□□□ □□□□ □□ force-unlock □□□ □□□□ □ □□□. □□□ □□□
□□ □□□ □□□ □□ □□□ □□□□ □□ □□□ □□□ □□ □□□ □□□□□ □□ □□
□ □□ □□□□ □□□. □□□ □□ □□ □□ □□ □□□ □□□□ □□□□ □□□□□ □□
□□ □□□ □ □□□□. □□□□ = [State Locking], [□□□□: force-unlock]

NEW QUESTION: 15

Terraform□ □□□ □□ □□□ □□□ □□ □□□ □□□□□.

- A. Terraform □□
- B. Terraform □□
- C. Terraform □□
- D. Terraform □□ □□

Answer: D (LEAVE A REPLY)

□□ □□ □□□□ Terraform□ □□ □□□□ □□□□ □□□□□ Terraform□ □□□ □□ □
□□ □□□ □□ □□□□□. -refresh=false □□□□ □□□□ □□ □ □□ □□□ □□□ □□
□□ □□ □□ □□□ □□ □□□□.

NEW QUESTION: 16

□□□ □□ □□ □□□ □□□ □□ □□□ □□□ □ □□□□.

- A. □
- B. □□

Answer: B (LEAVE A REPLY)

□□□ □□ □□□ □□□□□ □□□□ □□ □□□ □□□□ □□ □□□ □□ □□□□ □□
□□□ □□ □□ □□□ □□□ □□ □□□ □□□ □□ □□□□. □□□ □□ □□□ □□□□
□□ □ □□ □□□ □□□ □□ □□□□ □ □□□□.

Terraform-Associate-003 □□ □□□ □□□□□ □□ DumpTop □□ □□□□ □□□
Terraform-Associate-003 □□! DumpTop □ □□ **Terraform-Associate-003** □□ □□□ □□
□□□□, DumpTop Terraform-Associate-003 □□ □□□ □□□□□□□□□ □□□ □□□□
□□□. □□□□ □□□ □□□□ □□ DumpTop Terraform-Associate-003 □□□ □□□□
□. <https://www.dumptop.com/HashiCorp/Terraform-Associate-003-dump.html> (226 Q&As
Dumps, **30%OFF** Special Discount: **KrDump**)

NEW QUESTION: 17

□□

Terraform □□□ □□□ □□□□ □□ □□ □□□□ □□ □□ Terraform □□□ □□ □□□□
□ terraform fmt□ □□□□ □□□. □ □□□ □□□□ □□ □□ □□□ □□□ □□ Terraform
□□ □□□ □□□ □□ □□□ □□□□□. □□□ Terraform □□□□□□□ □□□□ □□□
□ □□□□□ □ □□□ □□□□ □□ □□□□. □ □□□ □□ □□□□ □□□□□ □□□ □
□ □□□ □□□ □□□ □□□ □□ □□□ □ □□□ □ □□□ □ □□□ □ □□□□. □□□□□
□□ = : □□: fmt : Terraform fmt □□□ □□□□ Terraform □□ □□ □□

NEW QUESTION: 21

Git□ □□□ □□□ □□□ □ v1.0.0 □□□ □□□□ □□ □□ □□□ □□□□□(□:
Git::https://example.com/vpc.git)?

- A. pref=v1.0.0 □□□ □□ □□□ □□□□□.
- B. □□ = "1.0.0" □□□□□ □□ □□□ □□□□□.
- C. GitHub□ □□□ □□□ □□ □□ 1.0.0□□ □□ □□□□ □□□□.

Answer: A (LEAVE A REPLY)

Git□ □□□ □□□ □□□ □ v1.0.0 □□□ □□□□ □□ □□ □□□ □□ □□□ ?ref=v1.0.0
□□□ □□□□ □□□□. □□ □□ □□ □□□ □□□ □ Terraform□ □□, □□ □□ □□□
□□ □□ Git □□□ □□□□□ □□□□□. □□ □□ □□ = "git::https://example.com/vpc.git?
ref=v1.0.0"□□□□. □□ □□ □□ □□□ □□□ □□□□ □□□□ □□ □□□□□ □□□□□.
□□ = [□□ □□], [□□ □□]

NEW QUESTION: 22

□□□□ □□□ Terraform □□□□ □□□□ VM(□□ □□)□ □□□□ □□□ □□□ □□ □
□□ □□□ □□□□□. □□ VM Terraform□ □□□□□ □□□□ □□ □□ VM ID □□□ □
□□□.

- □ Terraform□ □□□□ □□□□□ □□□□ □ □□□ □ □□ □□□ □□□□□?
- A. terraform state list□ □□□□ □□ VM□ □□□ □□ □□ □ VM□ □□ terraform state show
□ □□□□ Terraform□ □□□□ VM ID□ □□□□.
- B. □□ VM□ ID□ □□ □□□ □□□□□ □□□ □□□□□ □□ terraform □□□ □□□□ □
□□ □□□□□.
- C. □□ VM□□ Terraform taint/code□ □□□□ □□ □□□□□.
- D. Terraform □□ □□/□□□ □□□□ □□ □□□ □□□ ID□ □□□□□.

Answer: A (LEAVE A REPLY)

terraform state list □□□ □□ □□ □□1□□ Terraform□ □□□□ □□ □□□□ □□□□□.
terraform state show □□□ □□ file2□ □□ □□ □□□□ □□□ □□□□□. □ □ □□□ □
□□□ □□□ VM ID□ □□ □□□ VM ID□ □□□□ Terraform□□ □□□□ VM ID□ □□□
□ □□□□.

NEW QUESTION: 23

□□ □ □□□ Terraform □□ □□□ □□ □□ □□□□□?

Terraform is an open-source Infrastructure as Code (IaC) tool that allows you to define and provision infrastructure in a declarative manner. It uses HashiCorp Configuration Language (HCL) to describe the desired state of the infrastructure. Terraform can manage a wide variety of cloud and on-premises resources, including virtual machines, networks, storage, and more. It is designed to be multi-provider, allowing you to manage infrastructure across different cloud providers like AWS, Azure, and Google Cloud. Terraform also integrates with version control systems like Git and GitHub for collaboration and tracking changes.

It is developed by HashiCorp. Terraform is a multi-provider, open-source tool for managing infrastructure. It is used to define and provision infrastructure in a declarative manner.

NEW QUESTION: 27

gcloud is a command-line tool for interacting with Google Cloud Platform (GCP) resources. It is used to manage GCP resources like VMs, storage, and more. Terraform can be used to manage GCP resources in a declarative manner. The question asks how to import a VM into Terraform.

- A. Terraform Import-gcp
- B. gcloud VM Terraform
- C. gcloud VM terraform import
- D. gcloud VM Terraform VM

Answer: (SHOW ANSWER)

The correct answer is D. The command to import a VM into Terraform is `gcloud VM Terraform VM`. This command uses the `gcloud` CLI to interact with GCP resources and the `Terraform` CLI to manage infrastructure. The `import-gcp` command is not a valid Terraform command. The `gcloud VM terraform import` command is also not valid. The `gcloud VM Terraform` command is the correct way to import a VM into Terraform.

NEW QUESTION: 28

Terraform can be used to manage infrastructure in a declarative manner. The question asks how to manage AWS KMS keys using Terraform.

- A. Terraform
- B. gcloud
- C. -var
- D. gcloud

Answer: D (LEAVE A REPLY)

The correct answer is D. The command to manage AWS KMS keys using Terraform is `gcloud`. This command is used to interact with GCP resources, but it can also be used to manage AWS KMS keys. The `terraform` command is used to manage infrastructure in a declarative manner. The `-var` flag is used to pass variables to Terraform. The `gcloud` command is the correct way to manage AWS KMS keys using Terraform.

Terraform-Associate-003 [DumpTop](https://www.dumptop.com/HashiCorp/Terraform-Associate-003-dump.html) (226 Q&As Dumps, **30%OFF Special Discount: KrDump**)

NEW QUESTION: 32

Terraform `terraform init` command does what?

- A. `terraform init` command initializes the Terraform configuration files.
- B. `terraform init` command initializes the Terraform provider plugins.
- C. `terraform init` command initializes the Terraform workspace.
- D. `terraform init` command initializes the Terraform state file.

Answer: B (LEAVE A REPLY)

The `terraform init` command is used to initialize the Terraform configuration files. It downloads the provider plugins and sets up the local Terraform environment. The `terraform init` command is used to initialize the Terraform workspace. The `terraform init` command is used to initialize the Terraform state file.

NEW QUESTION: 33

Terraform `terraform destroy` command does what? Terraform `terraform destroy` command destroys the infrastructure defined in the configuration files. Terraform `terraform destroy` command destroys the infrastructure defined in the configuration files. Terraform `terraform destroy` command destroys the infrastructure defined in the configuration files.

mil `terraform destroy` command destroys the infrastructure defined in the configuration files. Terraform `terraform destroy` command destroys the infrastructure defined in the configuration files.

- A. Terraform `rm -rf` command
- B. `terraform show :destroy` command
- C. `terraform destroy` command destroys the infrastructure defined in the configuration files.
- D. Terraform `plan .destroy` command

Answer: C,D (LEAVE A REPLY)

Terraform `terraform destroy` command destroys the infrastructure defined in the configuration files. Terraform `terraform destroy` command destroys the infrastructure defined in the configuration files. Terraform `terraform destroy` command destroys the infrastructure defined in the configuration files. Terraform `terraform destroy` command destroys the infrastructure defined in the configuration files. Terraform `terraform destroy` command destroys the infrastructure defined in the configuration files.

NEW QUESTION: 34

_____ `terraform init` command does what?

- A. `terraform init` command

- B. Terraform `terraform destroy` command is used to destroy the infrastructure.
- C. Terraform `terraform destroy` command is used to destroy the infrastructure.
- D. Terraform `tf destroy` command is used to destroy the infrastructure.
- E. Terraform `terraform destroy` command is used to destroy the infrastructure.

Answer: [\(SHOW ANSWER\)](#)

`terraform destroy` command is used to destroy the infrastructure. The `-target` option is used to specify the resource to destroy.

NEW QUESTION: 38

Terraform `terraform destroy` command is used to destroy the infrastructure.

- A. Terraform `terraform destroy` command is used to destroy the infrastructure.
- B. Terraform `terraform destroy` command is used to destroy the infrastructure.
- C. Terraform `terraform destroy` command is used to destroy the infrastructure.
- D. Terraform `terraform destroy` command is used to destroy the infrastructure.

Answer: D [\(LEAVE A REPLY\)](#)

Terraform `terraform destroy` command is used to destroy the infrastructure. The `-target` option is used to specify the resource to destroy. (Source: GitHub and GitLab) Terraform `terraform destroy` command is used to destroy the infrastructure.

NEW QUESTION: 39

Terraform `terraform destroy` command is used to destroy the infrastructure.

- A. `terraform destroy`
- B. ID
- C. `terraform destroy`
- D. `terraform destroy`

Answer: A [\(LEAVE A REPLY\)](#)

Terraform `terraform destroy` command is used to destroy the infrastructure.

NEW QUESTION: 40

DevOps `terraform destroy` command is used to destroy the infrastructure.

- A. `terraform destroy`
- B. Amazon S3
- C. Terraform `terraform destroy`
- D. `terraform destroy`

Answer: D ([LEAVE A REPLY](#))

□□

□□ □□ □□□ □□ □□ □□ □□□□ □□□□ Terraform□ □□□ □□□□ □□□
□4. □□ □□□ □□ □□□ □□ □□□ □□□ □ □□ □□□ □□□□□□.

NEW QUESTION: 41

□/□ □□ □□□□□ □□ Terraform □□□ □□□ □□□□ □□□?

- A. □□
- B. □□
- C. □□
- D. □□

Answer: B ([LEAVE A REPLY](#))

□/□ □□ □□□□ □ □□□□ □□ Terraform □□□ □□□ □□□□. □□ □□□ □□ □□
□ □□□□ □□ □□□□□ □□ □□□□□. □□ □□ □□ □□□ □ □ □□□ □□ □ □□
□ □□□□ □□□. □□ □□ var = { key1 = "value1", key2 = "value2" }□ □ □□ □/□ □□ □
□ □□□□. □□ □□ □□□□ □□□□□□ □□ □□ □□ □□□□□ □ □□□□□. □□ =
[□□□ □□], [□□ □□ □□]

NEW QUESTION: 42

□□ □ Terraform□□ □□□□ □□□□ □□ □□□ □□□□□ □□□□□ □ □□□□ □□
□ □□□□□?

- A. Terraform □□ □□ ' □□□_□□_□□
- B. Terraform □□ □□
- C. Terraform□ □□□_□□_□□□ □□□□.
- D. Terraform □□ □□ □□□_□□_□□

Answer: ([SHOW ANSWER](#))

□□ □□□ □□□ □□□ □□□□ Terraform□□ □□□□ □□□□ □□ □□□ □□ □□□
□□□□□ □ □□□□ □□□□□. □□ □□, terraform state show 'aws_instance.example'□
example□□□ AWS □□□□□ □□ □□ □□□ □□□□□.

NEW QUESTION: 43

□□ □ Terraform Cloud□ □□□ □□□ □□ □□□□□? □ □□ □□□ □□□□□.

- A. □□□□ □□□ □□ □□□
- B. □□ □□
- C. □ □□ □□□ □□□□□(UI)
- D. □□ □□ □□

Answer: C,D ([LEAVE A REPLY](#))

□□

□□ Terraform □□□ □□□ □□ □ □□ UI, □□ □□ □□□, □□ □□, □□ □□ □□ □□
□□ □□□ □□□□ Terraform Cloud□ □□□□□.

laC(Infrastructure as Code) is a paradigm that allows you to define and provision infrastructure using code. It is a declarative approach where you describe the desired state of your infrastructure, and the tool automatically provisions and manages it. This approach is more consistent and repeatable than manual configuration. Terraform is a popular tool for implementing IaC. It uses a Domain Specific Language (DSL) called HCL to define infrastructure. Terraform can manage a wide range of providers, including AWS, Azure, and Google Cloud. REST APIs are used by Terraform to interact with cloud providers. Terraform uses a state file to track the current state of the infrastructure. The state file is stored in a local directory or a remote backend. Terraform uses the state file to determine what changes need to be made to reach the desired state. Terraform uses a lock to prevent concurrent updates to the state file. Terraform uses a plan to show the changes that will be made. Terraform uses an apply to make the changes. Terraform uses a refresh to update the state file with the current state of the infrastructure. Terraform uses a destroy to remove the infrastructure. Terraform uses a workspace to manage multiple configurations. Terraform uses a provider to interact with the cloud providers. Terraform uses a resource to define the infrastructure. Terraform uses a module to reuse configurations. Terraform uses a variable to define configuration values. Terraform uses a provider plugin to interact with the cloud providers. Terraform uses a resource plugin to define the infrastructure. Terraform uses a module plugin to reuse configurations. Terraform uses a variable plugin to define configuration values. Terraform uses a provider plugin to interact with the cloud providers. Terraform uses a resource plugin to define the infrastructure. Terraform uses a module plugin to reuse configurations. Terraform uses a variable plugin to define configuration values.

NEW QUESTION: 50

Terraform can connect to a cloud provider using which of the following protocols?

- A. SSH
- B. HTTP
- C. REST
- D. Terraform CLI
- E. gRPC

Answer: E (LEAVE A REPLY)

Terraform uses gRPC to connect to cloud providers. gRPC is a high-performance, open-source, and cross-platform Remote Procedure Call (RPC) framework. It is based on HTTP/2 and uses Protocol Buffers for data interchange. Terraform uses gRPC to communicate with the Terraform CLI and the Terraform provider plugins. The Terraform CLI uses gRPC to communicate with the Terraform provider plugins. The Terraform provider plugins use gRPC to communicate with the cloud providers. Terraform uses terraform.tfstate to store the state of the infrastructure. Terraform uses terraform.tfstate.lock to lock the state file. Terraform uses terraform.tfstate.backup to backup the state file. Terraform uses terraform.tfstate.backup.lock to lock the backup state file. Terraform uses terraform.tfstate.backup.lock to lock the backup state file.

NEW QUESTION: 51

Which of the following is a valid Terraform command to upgrade a provider?

- A. Terraform providers -upgrade
- B. Terraform providers -update
- C. Terraform providers -refresh
- D. Terraform providers -install

Answer: C (LEAVE A REPLY)

Terraform uses terraform providers -refresh to upgrade a provider. The terraform providers -refresh command refreshes the state of the infrastructure and updates the provider version. The terraform providers -refresh command uses the terraform providers -refresh command to refresh the state of the infrastructure and update the provider version. The terraform providers -refresh command uses the terraform providers -refresh command to refresh the state of the infrastructure and update the provider version.

NEW QUESTION: 52

Which of the following is a valid Terraform command to initialize a new workspace?

- A. terraform workspace new
- B. Terraform workspace create
- C. terraform workspace init
- D. terraform workspace init

Answer: D (LEAVE A REPLY)

Terraform uses terraform workspace init to initialize a new workspace. The terraform workspace init command initializes a new workspace. The terraform workspace init command uses the terraform workspace init command to initialize a new workspace. The terraform workspace init command uses the terraform workspace init command to initialize a new workspace. The terraform workspace init command uses the terraform workspace init command to initialize a new workspace.


```

terraform {
  backend "s3" {
    bucket     = "terraform-state-prod"
    key        = "network/terraform.tfstate"
    region     = "us-east-1"
    access_key = "AKIAIOSFODNN7EXAMPLE"
    secret_key = "wJalrXUtnFEMI/K7MDENG/bPxRf1CYEXAMPLEKEY"
  }

  required_providers {
    aws = {
      source = "hashicorp/aws"
      version = "~> 3.38"
    }
  }

  required_version = ">= 0.15"
}

```



- A. Terraform uses the AWS CLI to connect to S3.
- B. Terraform uses the AWS CLI to connect to S3.
- C. Terraform uses the AWS CLI to connect to S3.
- D. Terraform uses the AWS CLI to connect to S3.

Answer: (SHOW ANSWER)

Terraform uses the AWS CLI to connect to S3.

NEW QUESTION: 82

- Terraform uses the AWS CLI to connect to S3.
- A. Terraform uses the AWS CLI to connect to S3.
 - B. Terraform uses the AWS CLI to connect to S3.
 - C. Terraform uses the AWS CLI to connect to S3.
 - D. Terraform uses the AWS CLI to connect to S3.

Answer: D (LEAVE A REPLY)

Terraform uses the AWS CLI to connect to S3.

NEW QUESTION: 83

- A.
- B.
- C.
- D.

Answer: ([SHOW ANSWER](#))

Terraform `provider "aws" {`, `region = "us-east-1"`, `profile = "my-profile"` Terraform `provider "aws" {` `region = "us-east-1"`. Terraform `provider "aws" {` `region = "us-east-1"` `profile = "my-profile"` `region = "us-east-1"` `profile = "my-profile"`.

NEW QUESTION: 86

Terraform `provider "aws" {` `region = "us-east-1"`?

- A. Terraform `provider "aws" {` `region = "us-east-1"`
- B. Terraform `provider "aws" {` `region = "us-east-1"`
- C. Terraform `provider "aws" {` `region = "us-east-1"`
- D. `provider "aws" {` `region = "us-east-1"`
- E. `provider "aws" {` `region = "us-east-1"`

Answer: ([SHOW ANSWER](#))

Terraform `provider "aws" {` `region = "us-east-1"` Terraform `provider "aws" {` `region = "us-east-1"`. Terraform `provider "aws" {` `region = "us-east-1"` ID `provider "aws" {` `region = "us-east-1"` `profile = "my-profile"` `region = "us-east-1"`. `provider "aws" {` `region = "us-east-1"` `profile = "my-profile"` `region = "us-east-1"` `profile = "my-profile"`.

NEW QUESTION: 87

Splat(*) `provider "aws" {` `region = "us-east-1"` `profile = "my-profile"` `region = "us-east-1"`.

- A.
- B.

Answer: ([SHOW ANSWER](#))

`provider "aws" {` `region = "us-east-1"` `profile = "my-profile"` `region = "us-east-1"` `profile = "my-profile"` `region = "us-east-1"` `profile = "my-profile"`. Terraform `provider "aws" {` `region = "us-east-1"` `profile = "my-profile"` `region = "us-east-1"` `profile = "my-profile"`.

NEW QUESTION: 88

`provider "aws" {` `region = "us-east-1"` Terraform `provider "aws" {` `region = "us-east-1"` `profile = "my-profile"` `region = "us-east-1"` `profile = "my-profile"`?

- A. `provider "aws" {` `region = "us-east-1"`
- B. `provider "aws" {` `region = "us-east-1"`
- C. `provider "aws" {` `region = "us-east-1"` `profile = "my-profile"`
- D. `provider "aws" {` `region = "us-east-1"`

Answer: A ([LEAVE A REPLY](#))

NEW QUESTION: 94

Which of the following is a valid Terraform configuration for a provider? (Select one)

- A. provider "aws" {}
- B. provider "aws" { region = "us-east-1" }
- C. provider "aws" { region = "us-east-1" }
- D. provider "aws" { region = "us-east-1" }

Answer: A ([LEAVE A REPLY](#))

The correct answer is A. The provider block in Terraform is used to configure the providers that Terraform will use to manage the infrastructure. The provider block is defined in the configuration file and is used to specify the provider name and any configuration options. In this case, the provider "aws" is configured with no options, which is a valid configuration.

NEW QUESTION: 95

Which of the following is a valid Terraform configuration for a provider? (Select one)

- A. provider "aws" {}
- B. provider "aws" { region = "us-east-1" }
- C. provider "aws" Terraform {}
- D. provider "aws" {}

Answer: ([SHOW ANSWER](#))

The correct answer is A. The provider block in Terraform is used to configure the providers that Terraform will use to manage the infrastructure. The provider block is defined in the configuration file and is used to specify the provider name and any configuration options. In this case, the provider "aws" is configured with no options, which is a valid configuration. The other options are invalid because they either use the wrong provider name or include invalid configuration options.

NEW QUESTION: 96

Which of the following is a valid Terraform configuration for a provider? (Select one)

- A. provider "aws" {}
- B. provider secrets.tfvars {}
- C. Terraform {}

B.

```
{
  name = John
  age = fifty two
}
```

C.

```
{
  name = John
  age = "52"
}
```

D.

```
{
  name = "John"
  age = 52
}
```

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 100

VCS(□□ □□ □□□) □□□□ □□□ Terraform Cloud □□□□□□ □□□ □□□ □□□□ □□□□?

A. Terraform Cloud □□ □□□□ VCS □□□ □□□□□□ □□□□ □□□ □□□ □ □□□ □.

B. Terraform Cloud □□□□□ □□□ VCS □□ □□□□ □ □□□ □□ □□ □□□ □□□□ □.

C. Terraform Cloud □□ □□□□ VCS □□ □□□□□□ □□□ □□□ □ □□□□.

D. VCS □□□ □□□□ Terraform Cloud □□□□□ □□□ □□□□ □□ □ □□□ □ □ □□ □□.

Answer: B ([LEAVE A REPLY](#))

□□

□□□ Terraform Cloud □□□□□□ □□□ □□□□□ VCS □□□□ Terraform □□ □□□ □□ □□□ □□□□ □□ □□ □ □□ □□□ □□□□□.

NEW QUESTION: 101

□□ terraform init□ □□□□ □□□□?

A. Terraform□ □□□ □□□ □□

B. □ Terraform □□□□ □□□ □□□□ □

C. □ Terraform □□□□□□ □□□ Terraform □□□ □□□ □□ Terraform □□□ □□□□ □

D. □ Terraform □□□□ □□□ □□□ □□ □□□□ Terraform □□□ □□□□ □□□□.

Answer: ([SHOW ANSWER](#))

□ Terraform □□□□ □□□ □□□ □□ □□□□ terraform □□□ □□□□ □□ terraform init □ □□□□ □□□. □ □□□ □□□ □□□□ □□□ □□□□□□, □□ □□ □□□ □□□□, □□ □□□ □□□ □□□□ □□ □□□□□ □□□□□□□□. □□ = : Terraform □□□□ □□□□

NEW QUESTION: 102

_____ □ Terraform□ □□□□ □□, □□□□ □□ □□□□ □□□ □□□□□□.

Terraform-Associate-003 ☐☐ ☐☐☐ ☐☐☐☐☐ ☐☐ DumpTop ☐☐ ☐☐☐☐ ☐☐☐
Terraform-Associate-003 ☐☐! DumpTop ☐ ☐☐ **Terraform-Associate-003** ☐☐ ☐☐☐ ☐☐
☐☐☐☐, DumpTop Terraform-Associate-003 ☐☐ ☐☐☐ ☐☐☐☐☐☐☐☐☐ ☐☐☐ ☐☐☐☐
☐☐☐. ☐☐☐☐ ☐☐☐ ☐☐☐☐ ☐☐ DumpTop Terraform-Associate-003 ☐☐☐ ☐☐☐☐
☐. <https://www.dumptop.com/HashiCorp/Terraform-Associate-003-dump.html> (226 Q&As
Dumps, **30%OFF Special Discount: KrDump**)