CERTSinside

Free Questions for TA-002-P by certsinside

Shared by Marks on 12-12-2023

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: **MultipleChoice**

What is a key benefit of the Terraform state file?

Options:

A- A state file can schedule recurring infrastructure tasks

B- A state file is a source of truth for resources provisioned with Terraform

C- A state file is a source of truth for resources provisioned with a public cloud console

D- A state file is the desired state expressed by the Terraform code files

This is a key benefit of the Terraform state file, as it stores and tracks the metadata and attributes of the resources that are managed by Terraform, and allows Terraform to compare the current state with the desired state expressed by your configuration files.

Answer:

B

Question 2

Question Type: MultipleChoice

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?

Options:

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
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 runterraform init -upgradeto update them.

Answer:

A

Question 3

Question Type: MultipleChoice

Where does the Terraform local backend store its state?

Options:

A- In the terraform file

B- In the /tmp directory

C- In the terraform,tfstate file

D- In the user's terraform,state file

This is where the Terraform local backend stores its state, by default, unless you specify a different file name or location in your configuration. The local backend is the simplest backend type that stores the state file on your local disk.

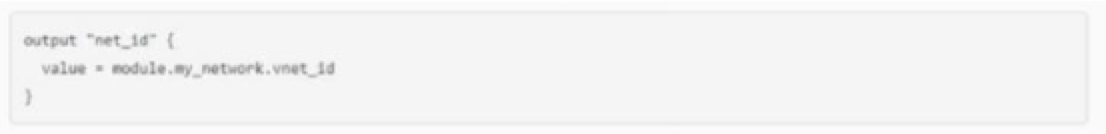
Answer:

C

Question 4

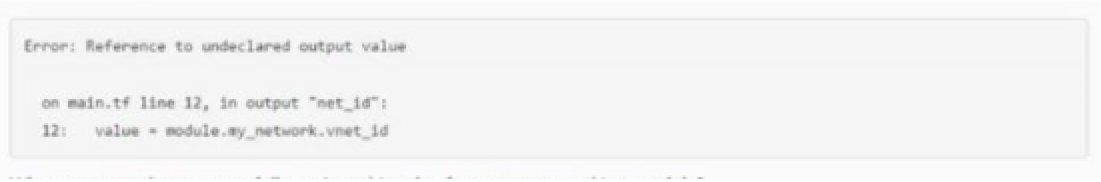
Question Type: MultipleChoice

You are using a networking module in your Terraform configuration with the name label my-network. In your main configuration you have the following code:



```
output "net_id" {
  value = module.my_network.vnet_id
}
```

When you run terraform validate, you get the following error:



```
Error: Reference to undeclared output value

  on main.tf line 12, in output "net_id":
  12:   value = module.my_network.vnet_id
```

What must you do to successfully retrieve this value from your networking module?

Options:

A- Change the reference value to my-network,outputs,vmet_id

B- Define the attribute vmet_id as a variable in the networking modeule

C- Define the attribute vnet_id as an output in the networking module

D- Change the reference value module.my,network,outputs,vnet_id

This is what you must do to successfully retrieve this value from your networking module, as it will expose the attribute as an output value that can be referenced by other modules or resources. The error message indicates that the networking module does not have an output value namedvnet_id, which causes the reference to fail.

Answer:

C

Question 5

Question Type: MultipleChoice

What type of block is used to construct a collection of nested configuration blocks?

Options:

A- Dynamic

B- For_each

C- Nesting

D- repeated.

This is the type of block that is used to construct a collection of nested configuration blocks, by using afor_eachargument to iterate over a collection value and generate a nested block for each element. For example, you can use a dynamic block to create multiple ingress rules for a security group resource.

Answer:

A

Question 6

Question Type: **MultipleChoice**

You much initialize your working directory before running terraform validate.

Options:

A- True

B- False

You must initialize your working directory before runningterraform validate, as it will ensure that all the required plugins and modules are installed and configured properly. If you skip this step, you may encounter errors or inconsistencies when validating your configuration files.

Answer:

A

Question 7

Question Type: MultipleChoice

A module can always refer to all variables declared in its parent module.

Options:

A- True

B- False

A module cannot always refer to all variables declared in its parent module, as it needs to explicitly declare input variables and assign values to them from the parent module's arguments. A module cannot access the parent module's variables directly, unless they are passed as input arguments.

Answer:

B

Question 8

Question Type: MultipleChoice

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

Options:

A- Tree

B- Map

C- List

D- set

This is not a valid Terraform collection type, as Terraform only supports three collection types: list, map, and set. A tree is a data structure that consists of nodes with parent-child relationships, which is not supported by Terraform.

Answer:

A

Question 9

Question Type: MultipleChoice

You want to know from which paths Terraform is loading providers referenced in your Terraform configuration (* files). You need to enable additional logging messages to find this out. Which of the following would achieve this?

Options:

A- Set verbose for each provider in your Terraform configuration

B- Set the environment variable TF_LOG_TRACE

C- Set the environment variable TF_LOG_PATH

D- Set the environment variable TF_log_TRACE

This will enable additional logging messages to find out from which paths Terraform is loading providers referenced in your Terraform configuration files, as it will set the log level toTRACE, which is the most verbose and detailed level.

Answer:

B

Question 10

Question Type: **MultipleChoice**

What Terraform command always causes a state file to be updated with changes that might have been made outside of Terraform?

Options:

A- Terraform plan --refresh-only

B- Terraform show --json

C- Terraform apply --lock-false

D- Terraform plan target-state

This is the command that always causes a state file to be updated with changes that might have been made outside of Terraform, as it will only refresh the state file with the current status of the real resources, without making any changes to them or creating a plan.

Answer:

A

Question 11

Question Type: **MultipleChoice**

How does Terraform determine dependencies between resources?

Options:

A- Terraform requires resource dependencies to be defined as modules and sourced in order

B- Terraform automatically builds a resource graph based on resources provisioners, special meta-parameters, and the stale file (if present}

C- Terraform requires resources in a configuration to be listed m the order they will be created to determine dependencies

D- Terraform requires all dependencies between resources to be specified using the depends_on parameter
This is how Terraform determines dependencies between resources, by using the references between them in the configuration files and other factors that affect the order of operations.

Answer:

B

Question 12

Question Type: **MultipleChoice**

The__________determines how Terraform creates, updates, or delete resources.

Options:

A- Terraform configuration

B- Terraform provisioner

C- Terraform provider

D- Terraform core

This is what determines how Terraform creates, updates, or deletes resources, as it is responsible for understanding API interactions with some service and exposing resources and data sources based on that API.

Answer:

C