HashiCorp

Exam Questions TA-002-P

HashiCorp Certified: Terraform Associate

NEW QUESTION 1
- (Exam Topic 1)
Which of the following is available only in Terraform Enterprise or Cloud workspaces and not in Terraform CLI?

A. Secure variable storage
B. Support for multiple cloud providers
C. Dry runs with terraform plan
D. Using the workspace as a data source

Answer: A

Explanation:

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

NEW QUESTION 2
- (Exam Topic 1)
What is the provider for this fictitious resource?

```
resource "aws_vpc" "main" {
    name = "test"
}
```

A. vpc
B. main
C. aws
D. test

Answer: C

Explanation:

Reference: https://docs.aws.amazon.com/cloudformation-cli/latest/userguide/resource-types.html

NEW QUESTION 3
- (Exam Topic 1)
A fellow developer on your team is asking for some help in refactoring their Terraform code. As part of their application's architecture, they are going to tear down an existing deployment managed by Terraform and deploy new. However, there is a server resource named aws_instance.ubuntu[1] they would like to keep to perform some additional analysis.
What command should be used to tell Terraform to no longer manage the resource?

A. terraform apply rm aws_instance.ubuntu[1]
B. terraform state rm aws_instance.ubuntu[1]
C. terraform plan rm aws_instance.ubuntu[1]
D. terraform delete aws_instance.ubuntu[1]

Answer: B

Explanation:
"You can use terraform state rm in the less common situation where you wish to remove a binding to an existing remote object without first destroying it, which will effectively make Terraform "forget" the object while it continues to exist in the remote system." https://www.terraform.io/cli/commands/state/rm

NEW QUESTION 4
- (Exam Topic 1)
You write a new Terraform configuration and immediately run terraform apply in the CLI using the local backend.
Why will the apply fail?

A. Terraform needs you to format your code according to best practices first
B. Terraform needs to install the necessary plugins first
C. The Terraform CLI needs you to log into Terraform cloud first
D. Terraform requires you to manually run terraform plan first

Answer: B

NEW QUESTION 5
- (Exam Topic 1)
Which of the following is not a valid string function in Terraform?

A. split
B. join
C. slice
D. chomp

Answer: C

Explanation:

https://www.terraform.io/language/functions

NEW QUESTION 6
- (Exam Topic 1)
Terraform provisioners can be added to any resource block.

A. True
B. False

Answer: A

Explanation:
https://www.phillipsj.net/posts/introduction-to-terraform-provisioners/
As you continue learning about Terraform, you will start hearing about provisioners. Terraform provisioners can be created on any resource and provide a way to execute actions on local or remote machines.
https://www.terraform.io/language/resources/provisioners/local-exec

NEW QUESTION 7
- (Exam Topic 1)
You need to deploy resources into two different cloud regions in the same Terraform configuration. To do that, you declare multiple provider configurations as follows:

```
provider "aws" {
  region = "us-east-1"
}

provider "aws" {
  alias = "west"
  region = "us-west-2"
}
```

What meta-argument do you need to configure in a resource block to deploy the resource to the "us-west-2" AWS region?

A. alias = west
B. provider = west
C. provider = aws.west
D. alias = aws.west

Answer: C

Explanation:

https://www.terraform.io/language/providers/configuration

NEW QUESTION 8
- (Exam Topic 1)
What is the name assigned by Terraform to reference this resource?

```
resource "azurerm_resource_group" "dev" {
  name = "test"
  location = "westus"
}
```

A. dev
B. azurerm_resource_group
C. azurerm
D. test

Answer: A

NEW QUESTION 9
- (Exam Topic 1)
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

NEW QUESTION 10
- (Exam Topic 1)
You should store secret data in the same version control repository as your Terraform configuration.

A. True
B. False

Answer: B

Explanation:

Reference: https://blog.gruntwork.io/a-comprehensive-guide-to-managing-secrets-in-your-terraform-code- 1d586955ace1

NEW QUESTION 10
- (Exam Topic 1)
When running the command terraform taint against a managed resource you want to force recreation upon, Terraform will immediately destroy and recreate the resource.

A. True
B. False

Answer: B

Explanation:
"The terraform taint command informs Terraform that a particular object has become degraded or damaged. Terraform represents this by marking the object as "tainted" in the Terraform state, and Terraform will propose to replace it in the next plan you create." FYI - This command is deprecated. For Terraform v0.15.2 and later, we recommend using the -replace option with terraform apply instead. For Terraform v0.15.2 and later, we recommend using the -replace option with terraform apply to force Terraform to replace an object even though there are no configuration changes that would require it.
https://www.terraform.io/cli/commands/taint

NEW QUESTION 15
- (Exam Topic 1)
The terraform.tfstate file always matches your currently built infrastructure.

A. True
B. False

Answer: B

Explanation:
Reference: https://www.terraform.io/docs/language/state/index.html

NEW QUESTION 17
- (Exam Topic 1)
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
Reference: https://registry.terraform.io/providers/hashicorp/aws/latest/docs/resources/instance

NEW QUESTION 22
- (Exam Topic 1)
A provider configuration block is required in every Terraform configuration. Example:

```
provider "provider_name" {

    ...

}
```

A. True
B. False

Answer: B

Explanation:
Unlike many other objects in the Terraform language, a provider block may be omitted if its contents would otherwise be empty. Terraform assumes an empty default configuration for any provider that is not explicitly configured. https://www.terraform.io/language/providers/configuration

NEW QUESTION 24
- (Exam Topic 1)
Which statement describes a goal of infrastructure as code?

A. An abstraction from vendor specific APIs
B. Write once, run anywhere
C. A pipeline process to test and deliver software
D. The programmatic configuration of resources

Answer: D

Explanation:
The purpose of infrastructure as code is to enable developers or operations teams to automatically manage, monitor and provision resources, rather than manually configure discrete hardware devices and operating systems. Infrastructure as code is sometimes referred to as programmable or software-defined infrastructure.

NEW QUESTION 26
- (Exam Topic 1)
Only the user that generated a plan may apply it.

A. True
B. False

Answer: B

NEW QUESTION 29
- (Exam Topic 1)
Your security team scanned some Terraform workspaces and found secrets stored in a plaintext in state files. How can you protect sensitive data stored in Terraform state files?

A. Delete the state file every time you run Terraform
B. Store the state in an encrypted backend
C. Edit your state file to scrub out the sensitive data
D. Always store your secrets in a secrets.tfvars file.

Answer: B

NEW QUESTION 33
- (Exam Topic 1)
What features does the hosted service Terraform Cloud provide? (Choose two.)

A. Automated infrastructure deployment visualization
B. Automatic backups
C. Remote state storage
D. A web-based user interface (UI)

Answer: CD

Explanation:
https://www.terraform.io/enterprise/admin/infrastructure/backup-restore

NEW QUESTION 35
- (Exam Topic 1)
Terraform can import modules from a number of sources – which of the following is not a valid source?

A. FTP server
B. GitHub repository
C. Local path
D. Terraform Module Registry

Answer: A

Explanation:
https://www.terraform.io/language/modules/sources

NEW QUESTION 39
- (Exam Topic 1)
Which of the following is the correct way to pass the value in the variable num_servers into a module with the input servers?

A. servers = num_servers
B. servers = variable.num_servers
C. servers = var(num_servers)
D. servers = var.num_servers

Answer: D

Explanation:
"Within the module that declared a variable, its value can be accessed from within expressions as var.<NAME>, where <NAME> matches the label given in the declaration block:
Note: Input variables are created by a variable block, but you reference them as attributes on an object named var."
https://www.terraform.io/language/values/variables#using-input-variable-values

NEW QUESTION 43
- (Exam Topic 1)
What type of block is used to construct a collection of nested configuration blocks?

A. for_each
B. repeated
C. nesting
D. dynamic

Answer: D

Explanation:
 https://www.terraform.io/language/expressions/dynamic-blocks

NEW QUESTION 44
- (Exam Topic 1)
FILL BLANK
Which flag would you add to terraform plan to save the execution plan to a file?
Type your answer in the field provided. The text field is not case-sensitive and all variations of the correct answer are accepted.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
"You can use the optional -out=FILE option to save the generated plan to a file on disk, which you can later execute by passing the file to terraform apply as an extra argument. This two-step workflow is primarily intended for when running Terraform in automation. If you run terraform plan without the -out=FILE option then it will create a speculative plan, which is a description of the effect of the plan but without any intent to actually apply it." https://www.terraform.io/cli/commands/plan

NEW QUESTION 49
- (Exam Topic 1)
You have declared an input variable called environment in your parent module. What must you do to pass the value to a child module in the configuration?

A. Add node_count = var.node_count
B. Declare the variable in a terraform.tfvars file
C. Declare a node_count input variable for child module
D. Nothing, child modules inherit variables of parent module

Answer: C

Explanation:
"That module may call other modules and connect them together by passing output values from one to input values of another."
https://www.terraform.io/language/modules/develop

NEW QUESTION 52
- (Exam Topic 1)
Terraform variables and outputs that set the "description" argument will store that description in the state file.

A. True
B. False

Answer: B

Explanation:

Reference: https://www.terraform.io/docs/language/values/outputs.html

NEW QUESTION 54
- (Exam Topic 1)
A Terraform local value can reference other Terraform local values.

A. True
B. False

Answer: A

Explanation:
"The expressions in local values are not limited to literal constants; they can also reference other values in the module in order to transform or combine them, including variables, resource attributes, or other local values:" https://www.terraform.io/language/values/locals#declaring-a-local-value

NEW QUESTION 59
- (Exam Topic 1)
Setting the TF_LOG environment variable to DEBUG causes debug messages to be logged into syslog.

A. True
B. False

Answer: B

Explanation:
TF_LOG_PATH IS NOT REQUIRED, in the docs, they do not mention HAVE TO SET TF_LOG_PATH, it is optional, therefore without TF_LOG_PATH will cause detailed logs to appear on stderr.
https://www.computerhope.com/jargon/s/stderr.htm#:~:text=Stderr%2C%20also%20known%20as%20standard,

NEW QUESTION 63
- (Exam Topic 1)
What is the name assigned by Terraform to reference this resource?

```
mainresource "google_compute_instance" "main" {
    name = "test"
}
```

A. compute_instance
B. main
C. google
D. test

Answer: B

NEW QUESTION 66
- (Exam Topic 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: BC

NEW QUESTION 71
- (Exam Topic 1)
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.
Reference: https://www.terraform.io/docs/language/settings/backends/local.html

NEW QUESTION 73
- (Exam Topic 1)
Which of the following is not a valid Terraform collection type?

A. list

B. map
C. tree
D. set

Answer: C

Explanation:
https://www.terraform.io/language/expressions/type-constraints#collection-types

NEW QUESTION 76
- (Exam Topic 1)
If you manually destroy infrastructure, what is the best practice reflecting this change in Terraform?

A. Run terraform refresh
B. It will happen automatically
C. Manually update the state fire
D. Run terraform import

Answer: A

Explanation:
https://www.terraform.io/cli/commands/refresh#:~:text=The%20terraform%20refresh%20command%20reads%

NEW QUESTION 81
- (Exam Topic 1)
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

NEW QUESTION 82
- (Exam Topic 1)
Which backend does the Terraform CLI use by default?

A. Terraform Cloud
B. Consul
C. Remote
D. Local

Answer: D

Explanation:
"By default, Terraform implicitly uses a backend called local to store state as a local file on disk. Every other
backend stores state in a remote service of some kind, which allows multiple people to access it. Accessing state in a remote service generally requires some kind
of access credentials, since state data contains extremely sensitive information." https://www.terraform.io/language/settings/backends

NEW QUESTION 85
- (Exam Topic 2)
Which of the below are paid features of Terraform Cloud?

A. Full API Coverage
B. Secure variable Storage
C. Roles/ Team management
D. Cost Estimation
E. Private Module Registry
F. Sentinel policies

Answer: CDF

Explanation:
https://www.hashicorp.com/products/terraform/pricing/

NEW QUESTION 89
- (Exam Topic 2)
What is the standard workflow that a developer follows while working with terraform open source version?

A. Run terraform refresh to update the terraform state , then write the terraform code , and finally run terraform apply.
B. Run terraform destroy first since you need to start from fresh every time , before running terraform apply.
C. Write terraform code , and run terraform push , to update the terraform state to the remote repo , which in turn will take care of the next steps.

D. Write the terraform code on the developer machine , run terraform plan to check the changes , and run terraform apply to provision the infra.

Answer: D

Explanation:
You do not need to run terraform refresh as terraform plan implicitly will run terraform refresh. https://www.terraform.io/guides/core-workflow.html

NEW QUESTION 90
- (Exam Topic 2)
While using generic git repository as a module source, which of the below options allows terraform to select a specific version or tag instead of selecting the HEAD.

A. Append ref argument asmodule "vpc" { source = "git::https://example.com/vpc.git?ref=v1.2.0"}
B. Append version argument asmodule "vpc" { source = "git::https://example.com/vpc.git?version=v1.2.0"}
C. Append ref argument asmodule "vpc" { source = "git::https://example.com/vpc.git#ref=v1.2.0"}
D. By default, Terraform will clone and use the default branch (referenced by HEAD) in the selected repository and you can not override this.

Answer: A

Explanation:
By default, Terraform will clone and use the default branch (referenced by HEAD) in the selected repository. You can override this using the ref argument:
module "vpc" {
source = "git::https://example.com/vpc.git?ref=v1.2.0"
}
The value of the ref argument can be any reference that would be accepted by the git checkout command, including branch and tag names.
https://www.terraform.io/docs/modules/sources.html

NEW QUESTION 92
- (Exam Topic 2)
The Terraform language does not support user-defined functions, and so only the functions built in to the language are available for use.

A. False
B. True

Answer: B

Explanation:
https://www.terraform.io/docs/configuration/functions.html

NEW QUESTION 94
- (Exam Topic 2)
Matt wants to import a manually created EC2 instance into terraform so that he can manage the EC2 instance through terraform going forward. He has written the configuration file of the EC2 instance before importing it to Terraform. Following is the code:
resource "aws_instance" "matt_ec2" { ami = "ami-bg2640de" instance_type = "t2.micro" vpc_security_group_ids = ["sg-6ae7d613", "sg-53370035"] key_name = "mysecret" subnet_id =
"subnet-9e3cfbc5" }
The instance id of that EC2 instance is i-0260835eb7e9bd40 How he can import data of EC2 to state file?

A. terraform import aws_instance.id = i-0260835eb7e9bd40
B. terraform import i-0260835eb7e9bd40
C. terraform import aws_instance.i-0260835eb7e9bd40
D. terraform import aws_instance.matt_ec2 i-0260835eb7e9bd40

Answer: D

Explanation:
https://www.terraform.io/docs/import/usage.html

NEW QUESTION 96
- (Exam Topic 2)
Provisioners should only be used as a last resort.

A. False
B. True

Answer: B

Explanation:
Provisioners are a Last Resort
Terraform includes the concept of provisioners as a measure of pragmatism, knowing that there will always be certain behaviors that can't be directly represented in Terraform's declarative model.
However, they also add a considerable amount of complexity and uncertainty to Terraform usage. Firstly, Terraform cannot model the actions of provisioners as part of a plan because they can in principle take any action. Secondly, successful use of provisioners requires coordinating many more details than Terraform usage usually requires: direct network access to your servers, issuing Terraform credentials to log in, making sure that all of the necessary external software is installed, etc.
The following sections describe some situations which can be solved with provisioners in principle, but where better solutions are also available. We do not recommend using provisioners for any of the use-cases described in the following sections.
Even if your specific use-case is not described in the following sections, we still recommend attempting to solve it using other techniques first, and use provisioners only if there is no other option.

https://www.terraform.io/docs/provisioners/index.html

NEW QUESTION 99
- (Exam Topic 2)
John wants to use two different regions to deploy two different EC2 instances. He has specified two provider blocks in his providers.tf file.
provider "aws" { region = "us-east-1" } provider "aws" { region = "us-west-2" }
When he run terraform plan he encountered an error. How to fix this?

A. Use another provider version
B. Use alias for region = "us-west-2"
C. Use default keyword with region = "us-east-1"
D. It can not be fixed

Answer: B

NEW QUESTION 104
- (Exam Topic 2)
ABC Enterprise has recently tied up with multiple small organizations for exchanging database information. Due to this, the firewall rules are increasing and are more than 100 rules. This is leading firewall configuration file that is difficult to manage. What is the way this type of configuration can be managed easily?

A. Terraform Backends
B. Terraform Functions
C. Dynamic Blocks
D. Terraform Expression

Answer: C

NEW QUESTION 105
- (Exam Topic 2)
The terraform init command is always safe to run multiple times, to bring the working directory up to date with changes in the configuration. Though subsequent runs may give errors, this command will never delete your existing configuration or state.

A. False
B. True

Answer: B

Explanation:
https://www.terraform.io/docs/commands/init.html

NEW QUESTION 110
- (Exam Topic 2)
You have created 2 workspaces PROD and RQA. You have switched to RQA and provisioned RQA infrastructure from this workspace. Where is your state file stored?

A. terraform.tfstate.d
B. terraform.d
C. terraform.tfstate.RQA
D. terraform.tfstate

Answer: A

NEW QUESTION 115
- (Exam Topic 2)
lookup retrieves the value of a single element from which of the below data type?

A. map
B. set
C. string
D. list

Answer: A

Explanation:
https://www.terraform.io/docs/configuration/functions/lookup.html

NEW QUESTION 117
- (Exam Topic 2)
Terraform init can indeed be run only a few times, because, every time terraform init will initialize the project
, and download all plugins from the internet repository , regardless of whether they were present or not , and this increases the waiting time

A. True
B. False

Answer: B

Explanation:
Re-running init with modules already installed will install the sources for any modules that were added to configuration since the last init, but will not change any already-installed modules. Use -upgrade to override this behavior, updating all modules to the latest available source code.
https://www.terraform.io/docs/commands/init.html

NEW QUESTION 119
- (Exam Topic 2)
Which Terraform command will force a marked resource to be destroyed and recreated on the next apply?

A. terraform fmt
B. terraform destroy
C. terraform taint
D. terraform refresh

Answer: C

Explanation:
The terraform taint command manually marks a Terraform-managed resource as tainted, forcing it to be destroyed and recreated on the next apply.
This command will not modify infrastructure, but does modify the state file in order to mark a resource as tainted. Once a resource is marked as tainted, the next plan will show that the resource will be destroyed and recreated and the next apply will implement this change.
Forcing the recreation of a resource is useful when you want a certain side effect of recreation that is not visible in the attributes of a resource. For example: re-running provisioners will cause the node to be different or rebooting the machine from a base image will cause new startup scripts to run.
Note that tainting a resource for recreation may affect resources that depend on the newly tainted resource. For example, a DNS resource that uses the IP address of a server may need to be modified to reflect the potentially new IP address of a tainted server. The plan command will show this if this is the case.
https://www.terraform.io/docs/commands/taint.html

NEW QUESTION 123
- (Exam Topic 2)
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

NEW QUESTION 127
- (Exam Topic 2)
Which of the following Terraform files should be ignored by Git when committing code to a repo? (select Three)

A. Files named exactly terraform.tfvars or terraform.tfvars.json.
B. Any files with names ending in .auto.tfvars or .auto.tfvars.json.
C. input.tf
D. terraform.tfstate
E. output.tf

Answer: ABD

Explanation:
The .gitignore file should be configured to ignore Terraform files that either contain sensitive data or are not required to save.
Terraform state (terraform.tfstate) can contain sensitive data, depending on the resources in use and your definition of "sensitive." The state contains resource IDs and all resource attributes. For resources such as databases, this may contain initial passwords.
When using local state, state is stored in plain-text JSON files.
The terraform.tfvars file may contain sensitive data, such as passwords or IP addresses of an environment that you may not want to share with others.

NEW QUESTION 130
- (Exam Topic 2)
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.

NEW QUESTION 132
- (Exam Topic 2)
terraform refresh will update 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. If the state is changed, this may cause changes to occur during the next plan or apply.

NEW QUESTION 133
- (Exam Topic 2)
Terraform works well in Windows but a Windows server is required.

A. False
B. True

Answer: A

Explanation:
You may see this QUESTION NO: in actual exam. Please remember : Terraform does not require GO language to be installed as a prerequisite and it does not require a Windows Server as well.

NEW QUESTION 135
- (Exam Topic 2)
You have declared a variable name my_var in terraform configuration without a value associated with it. variable my_var {}
After running terraform plan it will show an error as variable is not defined.

A. True
B. False

Answer: B

Explanation:
Input variables are usually defined by stating a name, type and a default value. However, the type and default values are not strictly necessary. Terraform can deduct the type of the variable from the default or input value.
Variables can be predetermined in a file or included in the command-line options. As such, the simplest variable is just a name while the type and value are selected based on the input.
variable "variable_name" {}
terraform apply -var variable_name="value"
The input variables, like the one above, use a couple of different types: strings, lists, maps, and boolean. Here are some examples of how each type are defined and used.
String
Strings mark a single value per structure and are commonly used to simplify and make complicated values more user-friendly. Below is an example of a string variable definition.
variable "template" { type = string
default = "01000000-0000-4000-8000-000030080200"
}
A string variable can then be used in resource plans. Surrounded by double quotes, string variables are a simple substitution such as the example underneath.
storage = var.template List
Another type of Terraform variables lists. They work much like a numbered catalogue of values. Each value can be called by their corresponding index in the list. Here is an example of a list variable definition.
variable "users" { type = list
default = ["root", "user1", "user2"]
}
Lists can be used in the resource plans similarly to strings, but you'll also need to denote the index of the value you are looking for.
username = var.users[0] Map
Maps are a collection of string keys and string values. These can be useful for selecting values based on predefined parameters such as the server configuration by the monthly price.
variable "plans" { type = map default = {
"5USD" = "1xCPU-1GB" "10USD" = "1xCPU-2GB" "20USD" = "2xCPU-4GB"
}
}
You can access the right value by using the matching key. For example, the variable below would set the plan to "1xCPU-1GB".
plan = var.plans["5USD"]
The values matching to their keys can also be used to look up information in other maps. For example, underneath is a shortlist of plans and their corresponding storage sizes.
variable "storage_sizes" { type = map
default = {
"1xCPU-1GB" = "25"
"1xCPU-2GB" = "50"
"2xCPU-4GB" = "80"
}
}
These can then be used to find the right storage size based on the monthly price as defined in the previous example.
size = lookup(var.storage_sizes, var.plans["5USD"])
Boolean
The last of the available variable type is boolean. They give the option to employ simple true or false values. For example, you might wish to have a variable that decides when to generate the root user password on a new deployment.
variable "set_password" { default = false
}

The above example boolean can be used similarly to a string variable by simply marking down the correct variable.
create_password = var.set_password
By default, the value is set to false in this example. However, you can overwrite the variable at deployment by assigning a different value in a command-line variable.
terraform apply -var set_password="true"

NEW QUESTION 138
- (Exam Topic 2)
By default, a defined provisioner is a creation-time provisioner.

A. True
B. False

Answer: A

Explanation:
https://www.terraform.io/docs/provisioners/index.html

NEW QUESTION 141
- (Exam Topic 2)
Which of the following best describes a Terraform provider?

A. A plugin that Terraform uses to translate the API interactions with the service or provider.
B. Serves as a parameter for a Terraform module that allows a module to be customized.
C. Describes an infrastructure object, such as a virtual network, compute instance, or other components.
D. A container for multiple resources that are used together.

Answer: A

Explanation:
A provider is responsible for understanding API interactions and exposing resources. Providers generally are an IaaS (e.g. Alibaba Cloud, AWS, GCP, Microsoft Azure, OpenStack), PaaS (e.g. Heroku), or SaaS services (e.g.Terraform Cloud, DNSimple, Cloudflare).
https://www.terraform.io/docs/providers/index.html

NEW QUESTION 146
- (Exam Topic 2)
Which of the below configuration file formats are supported by Terraform? (Select TWO)

A. Node
B. JSON
C. Go
D. YAML
E. HCL

Answer: BE

Explanation:
Terraform supports both HashiCorp Configuration Language (HCL) and JSON formats for configurations. https://www.terraform.io/docs/configuration/

NEW QUESTION 151
- (Exam Topic 2)
Which of the following type of variable allows multiple values of several distinct types to be grouped together as a single value?

A. Map
B. Object
C. Tuple
D. List

Answer: BC

Explanation:
Structural type of variable allows multiple values of several distinct types to be grouped together as a single value. They require a schema as an argument, to specify which types are allowed for which elements.
https://www.terraform.io/docs/configuration/types.html

NEW QUESTION 153
- (Exam Topic 2)
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: CD

Explanation:
https://www.terraform.io/docs/commands/refresh.html

NEW QUESTION 155
- (Exam Topic 2)
You do not need to specify every required argument in the backend configuration. Omitting certain arguments may be desirable to avoid storing secrets, such as access keys, within the main configuration. When some or all of the arguments are omitted, we call this a _____.

A. First Time Configuration
B. Default Configuration
C. Changing Configuration
D. Partial Configuration
E. Incomplete Configuration

Answer: D

Explanation:
You do not need to specify every required argument in the backend configuration. Omitting certain arguments may be desirable to avoid storing secrets, such as access keys, within the main configuration. When some or all of the arguments are omitted, we call this a partial configuration.
With a partial configuration, the remaining configuration arguments must be provided as part of the initialization process. There are several ways to supply the remaining arguments:
* Interactively: Terraform will interactively ask you for the required values, unless interactive input is disabled. Terraform will not prompt for optional values.
* File: A configuration file may be specified via the init command line. To specify a file, use the
-backend-config=PATH option when running terraform init. If the file contains secrets it may be kept in a secure data store, such as Vault, in which case it must be downloaded to the local disk before running Terraform.
* Command-line key/value pairs: Key/value pairs can be specified via the init command line. Note that many shells retain command-line flags in a history file, so this isn't recommended for secrets. To specify a single key/value pair, use the -backend-config="KEY=VALUE" option when running terraform init.
https://www.terraform.io/docs/backends/config.html#partial-configuration

NEW QUESTION 157
- (Exam Topic 2)
What is the purpose of using the local-exec provisioner? (Select Two)

A. To invoke a local executable.
B. Executes a command on the resource to invoke an update to the Terraform state.
C. To execute one or more commands on the machine running Terraform.
D. Ensures that the resource is only executed in the local infrastructure where Terraform is deployed.

Answer: AC

Explanation:
The local-exec provisioner invokes a local executable after a resource is created. This invokes a process on the machine running Terraform, not on the resource. Note that even though the resource will be fully created when the provisioner is run, there is no guarantee that it will be in an operable state - for example system services such as sshd may not be started yet on compute resources.
Example usage
resource "aws_instance" "web" {
# ...
provisioner "local-exec" {
command = "echo ${aws_instance.web.private_ip} >> private_ips.txt"
}
}
Note: Provisioners should only be used as a last resort. For most common situations there are better alternatives.
https://www.terraform.io/docs/provisioners/local-exec.html

NEW QUESTION 162
- (Exam Topic 2)
What does terrafom plan do ?

A. Create an execution plan by evaluating the difference between configuration file and state file.
B. Performs a refresh, unless explicitly disabled, and then apply the changes that are necessary to achieve the desired state specified in the configuration files.
C. Create an execution plan by evaluating the difference between configuration file and actual infrastructure.
D. Checks whether the execution plan for a set of changes matches your expectations by making changes to real resources or to the state.

Answer: A

NEW QUESTION 164
- (Exam Topic 2)
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

NEW QUESTION 166
- (Exam Topic 3)
Eric needs to make use of module within his terraform code. Should the module always be public and open-source to be able to be used?

A. False
B. True

Answer: A

Explanation:
Terraform module need not be public and open-source. Module can be placed in
* Local paths
* Terraform Registry
* GitHub
* Bitbucket
* Generic Git, Mercurial repositories
* HTTP URLs
* S3 buckets
* GCS buckets https://www.terraform.io/docs/modules/sources.html

NEW QUESTION 167
- (Exam Topic 3)
The canonical format may change in minor ways between Terraform versions, so after upgrading Terraform it is recommended to proactively run.

A. terraform fmt
B. terraform init
C. terraform validate
D. terraform plan

Answer: A

NEW QUESTION 169
- (Exam Topic 3)
Once a resource is marked as tainted, the next plan will show that the resource will be _____ and _____ and the next apply will implement this change.

A. recreated and tainted
B. destroyed and not recreated
C. tainted and not destroyed
D. destroyed and recreated

Answer: D

NEW QUESTION 171
- (Exam Topic 3)
Your manager has instructed you to start using terraform for your day-to-day operations, but your security team is concerned about the terraform state files. They have heard it contains confidential information, and are worried that it will not be securely protected. What should be your response to the security team in this regard?

A. Inform the security team that using terraform state is optional . There are ways to avoid it , and you will do the same.
B. Ensure that the state is managed in a remote backend , preferably an enterprise grade state management system like Terraform Cloud.
C. Mask the confidential entries in the terraform state file , using Vault Enterprise, another Hashicorp product , while keeping it locally.
D. Keep the state file locally on each developer machine , and ensure that there is a local protection software like KeyPass protecting it.

Answer: B

Explanation:
https://www.terraform.io/docs/state/index.html
State is very important topic for exam. Please read all of the below subtopics Purpose
Import Existing Resources Locking
Workspaces Remote State Sensitive Data

NEW QUESTION 174
- (Exam Topic 3)
After running into issues with Terraform, you need to enable verbose logging to assist with troubleshooting the error. Which of the following values provides the MOST verbose logging?

A. ERROR
B. INFO
C. WARN
D. TRACE
E. DEBUG

Answer: D

Explanation:
Terraform has detailed logs that can be enabled by setting the TF_LOG environment variable to any value. This will cause detailed logs to appear on stderr.
You can set TF_LOG to one of the log levels TRACE, DEBUG, INFO, WARN or ERROR to change the verbosity of the logs. TRACE is the most verbose and it is the default if TF_LOG is set to something other than a log level name.
Examples:
export TF_LOG=DEBUG export TF_LOG=TRACE

NEW QUESTION 175
- (Exam Topic 3)
After creating a new workspace "PROD" you need to run the command terraform select PROD to switch to it.

A. False
B. True

Answer: A

Explanation:
By default, when you create a new workspace you are automatically switched to it
To create a new workspace and switch to it, you can use terraform workspace new <new_workspace_name>; to switch to a existing workspace you can use terraform workspace select <existing_workspace_name>;
Example:
$ terraform workspace new example
Created and switched to workspace "example"!
You're now on a new, empty workspace. Workspaces isolate their state, so if you run "terraform plan" Terraform will not see any existing state for this configuration.

NEW QUESTION 180
- (Exam Topic 3)
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: AC

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

NEW QUESTION 181
- (Exam Topic 3)
When multiple engineers start deploying infrastructure using the same state file, what is a feature of remote state storage that is critical to ensure the state doesn't become corrupt?

A. Object Storage
B. State Locking
C. WorkSpaces
D. Encryption

Answer: B

Explanation:
If supported by your backend, Terraform will lock your state for all operations that could write state. This prevents others from acquiring the lock and potentially corrupting your state.
State locking happens automatically on all operations that could write state. You won't see any message that it is happening. If state locking fails, Terraform will not continue. You can disable state locking for most commands with the -lock flag but it is not recommended.
If acquiring the lock is taking longer than expected, Terraform will output a status message. If Terraform doesn't output a message, state locking is still occurring if your backend supports it.
Not all backends support locking. Please view the list of backend types for details on whether a backend supports locking or not.
https://www.terraform.io/docs/state/locking.html

NEW QUESTION 182
- (Exam Topic 3)
You want terraform plan and terraform apply to be executed in Terraform Cloud's run environment but the output is to be streamed locally. Which one of the below you will choose?

A. Local Backends.
B. Terraform Backends.
C. This can be done using any of the local or remote backends.
D. Remote Backends.

Answer: D

Explanation:
When using full remote operations, operations like terraform plan or terraform apply can be executed in Terraform Cloud's run environment, with log output streaming to the local terminal. Remote plans and applies use variable values from the associated Terraform Cloud workspace.

Terraform Cloud can also be used with local operations, in which case only state is stored in the Terraform Cloud backend.
https://www.terraform.io/docs/backends/types/remote.html

NEW QUESTION 186
- (Exam Topic 3)
Which of the below command will upgrade the provider version to the latest acceptable one?

A. terraform plan upgrade
B. terraform provider -upgrade
C. terraform init -upgrade
D. terraform init -update

Answer: C

Explanation:
To upgrade to the latest acceptable version of each provider, run terraform init -upgrade. This command also upgrades to the latest versions of all Terraform modules.
https://www.terraform.io/docs/configuration/providers.html

NEW QUESTION 190
- (Exam Topic 3)
Which of the below commands will rename a EC2 instance without destroying and recreating it?

A. terraform state mv
B. terraform mv
C. terraform plan
D. terraform plan mv

Answer: A

NEW QUESTION 192
- (Exam Topic 3)
Which of the following variable definition files will terraform load automatically?

A. terraform.tfvar
B. Any files with names ending in .auto.tfvars.json
C. terraform.tfvars
D. terraform.tfvars.json

Answer: BCD

Explanation:
Terraform also automatically loads a number of variable definitions files if they are present: Files named exactly terraform.tfvars or terraform.tfvars.json.
Any files with names ending in .auto.tfvars or .auto.tfvars.json. https://www.terraform.io/docs/configuration/variables.html
https://www.terraform.io/docs/configuration/variables.html#variable-definitions-tfvars-files

NEW QUESTION 195
- (Exam Topic 3)
Terraform Enterprise currently supports running under which the following operating systems?

A. Ubuntu
B. Amazon Linux
C. Debian
D. CentOS
E. Red Hat Enterprise Linux
F. Oracle Linux

Answer: ABCDEF

Explanation:
Terraform Enterprise runs on Linux instances, and you must prepare a running Linux instance for Terraform Enterprise before running the installer. You will start and manage this instance like any other server.
Terraform Enterprise currently supports running under the following operating systems: Standalone deployment:
Debian 7.7+
Ubuntu 14.04.5 / 16.04 / 18.04
Red Hat Enterprise Linux 7.4 - 7.8 CentOS 6.x / 7.4 - 7.8
Amazon Linux 2014.03 / 2014.09 / 2015.03 / 2015.09 / 2016.03 / 2016.09 / 2017.03 / 2017.09 / 2018.03 / 2.0
Oracle Linux 7.4 - 7.8 https://www.terraform.io/docs/enterprise/before-installing/index.html

NEW QUESTION 197
- (Exam Topic 3)
You also have a defined the following environment variables in your shell: TF_itemNum =6, TF_VAR_itemNum =9. You also have a terraform.tfvars file with the following contents
itemNum = 7
When you run the following apply command, what is the value assigned to the itemNum variable? terraform apply -var itemNum =4

A. 10
B. 6

C. 1
D. 4
E. 3

Answer: D

Explanation:
The -var and -var-file methods of assigning variables have the highest precedence. https://www.terraform.io/docs/configuration/variables.html

NEW QUESTION 201
- (Exam Topic 3)
What kind of resource dependency is stored in terraform.tfstate file?

A. Both implicit and explicit dependencies are stored in state file.
B. Only explicit dependencies are stored in state file.
C. Only implicit dependencies are stored in state file.
D. No dependency information is stored in state file.

Answer: A

Explanation:
Terraform state captures all dependency information, both implicit and explicit. One purpose for state is to determine the proper order to destroy resources. When resources are created all of their dependency information is stored in the state. If you destroy a resource with dependencies, Terraform can still determine the correct destroy order for all other resources because the dependencies are stored in the state.
https://www.terraform.io/docs/state/purpose.html#metadata

NEW QUESTION 203
- (Exam Topic 3)
The terraform state command can be used to _____

A. Update current state
B. Refresh existing state file
C. Print the current state file in console
D. It is not a valid command

Answer: A

Explanation:
The terraform state command is used for advanced state management. Rather than modify the state directly, the terraform state commands can be used in many cases instead.
https://www.terraform.io/docs/commands/state/index.html

NEW QUESTION 205
- (Exam Topic 3)
You can migrate the Terraform backend but only if there are no resources currently being managed.

A. False
B. True

Answer: A

Explanation:
If you need to migrate to another backend, such as Terraform Cloud, so you can continue managing it. By migrating your Terraform state, you can hand off infrastructure without de-provisioning anything.
https://www.terraform.io/docs/cloud/migrate/index.html

NEW QUESTION 210
- (Exam Topic 3)
Hanah is writing a terraform configuration with nested modules, there are multiple places where she has to use the same conditional expression but she wants to avoid repeating the same values or expressions multiple times in the configuration,. What is a better approach to dealing with this?

A. Expressions
B. Local Values
C. Variables
D. Functions

Answer: B

Explanation:
https://www.terraform.io/docs/configuration/locals.html

NEW QUESTION 212
- (Exam Topic 3)
Multiple providers can be declared within a single Terraform configuration file.

A. True
B. False

Answer: A

Explanation:
You can optionally define multiple configurations for the same provider, and select which one to use on a per-resource or per-module basis. The primary reason for this is to support multiple regions for a cloud platform; other examples include targeting multiple Docker hosts, multiple Consul hosts, etc.
To include multiple configurations for a given provider, include multiple provider blocks with the same provider name, but set the alias meta-argument to an alias name to use for each additional configuration.
For Example
# The default provider configuration provider "aws" {
region = "us-east-1"
}
# Additional provider configuration for west coast region provider "aws" {
alias = "west" region = "us-west-2"
}
The provider block without alias set is known as the default provider configuration. When alias is set, it creates an additional provider configuration. For providers that have no required configuration arguments, the implied empty configuration is considered to be the default provider configuration.
https://www.terraform.io/docs/configuration/providers.html

NEW QUESTION 216
- (Exam Topic 3)
Why is it a good idea to declare the required version of a provider in a Terraform configuration file?
* 1. terraform
* 2. {
* 3. required_providers
* 4. {
* 5. aws = "~> 1.0"
* 6. }
* 7. }

A. To remove older versions of the provider.
B. To ensure that the provider version matches the version of Terraform you are using.
C. Providers are released on a separate schedule from Terraform itself; therefore a newer version could introduce breaking changes.
D. To match the version number of your application being deployed via Terraform.

Answer: C

NEW QUESTION 217
- (Exam Topic 3)
Which of the following are string functions? Select three

A. tostring
B. tonumber
C. Chomp
D. format
E. join

Answer: CDE

Explanation:
tonumber and tostring are Type Conversion function https://www.terraform.io/docs/configuration/functions.html

NEW QUESTION 221
- (Exam Topic 3)
A user has created three workspaces using the command line - prod, dev, and test. The user wants to create a fourth workspace named stage. Which command will the user execute to accomplish this?

A. terraform workspace new stage
B. terraform workspace -new stage
C. terraform workspace -create stage
D. terraform workspace create stage

Answer: A

Explanation:
The terraform workspace new command is used to create a new workspace. https://www.terraform.io/docs/commands/workspace/new.html

NEW QUESTION 224
- (Exam Topic 4)
A Terraform output that sets the ''sensitive'' argument to true will not store that value in the state file.

A. True
B. False

Answer: B

Explanation:
Reference: https://www.terraform.io/language/values/outputs

NEW QUESTION 227
- (Exam Topic 4)
How can a ticket-based system slow down infrastructure provisioning and limit the ability to scale? (Choose two.)

A. A full audit trail of the request and fulfillment process is generated
B. A request must be submitted for infrastructure changes
C. As additional resources are required, more tickets are submitted
D. A catalog of approved resources can be accessed from drop down lists in a request form

Answer: BC

NEW QUESTION 231
- (Exam Topic 4)
Select the answer below that completes the following statement: Terraform Cloud can be managed from the CLI but requires _____?

A. an API token
B. a TOTP token
C. a username and password
D. authentication using MFA

Answer: A

Explanation:
API and CLI access are managed with API tokens, which can be generated in the Terraform Cloud UI. Each user can generate any number of personal API tokens, which allow access with their own identity and permissions. Organizations and teams can also generate tokens for automating tasks that aren't tied to an individual user.

NEW QUESTION 236
- (Exam Topic 4)
As a developer, you want to ensure your plugins are up to date with the latest versions. Which Terraform command should you use?

A. terreform providers- upgrade
B. terreform apply -upgrade
C. terreform refresh -upgrade
D. terreformn Init -upgrade

Answer: D

NEW QUESTION 241
- (Exam Topic 4)
What does the command terraform fmt do?

A. Rewrite Terraform configuration files to a canonical format and style.
B. Deletes the existing configuration file.
C. Updates the font of the configuration file to the official font supported by HashiCorp.
D. Formats the state file in order to ensure the latest state of resources can be obtained.

Answer: A

Explanation:
The terraform fmt command is used to rewrite Terraform configuration files to a canonical format and style. This command applies a subset of the Terraform language style conventions, along with other minor adjustments for readability.
Other Terraform commands that generate Terraform configuration will produce configuration files that conform to the style imposed by terraform fmt, so using this style in your own files will ensure consistency.
https://www.terraform.io/docs/commands/fmt.html

NEW QUESTION 245
- (Exam Topic 4)
Terraform is currently being used by your organisation to create resources on AWS for the development of a web application. One of your coworkers wants to change the instance type to "t2.large" while keeping the default set values.
What adjustments does the teammate make in order to meet his goal?

A. Issue Terraform plan instance.type".t2.large" and it deploys the instance
B. Modify the tf.variableswith the instance type and issue terraform apply
C. Create a new file my.tfvars and add the type of the instance and issue terraform plan and apply
D. Modify the terraform.tfvars with the instance type and issue terraform plan and then terraform apply to deploy the instances

Answer: D

NEW QUESTION 248
- (Exam Topic 4)
In the example below, the depends_on argument creates what type of dependency?

A. implicit dependency
B. internal dependency
C. explicit dependency
D. non-dependency resource

Answer: C

NEW QUESTION 249
- (Exam Topic 4)
To check if all code in a Terraform configuration with multiple modules is properly formatted without making changes, what command should be run?

A. terraform fmt -check
B. terraform fmt -write-false
C. terraform fmt "list -recursive
D. terraform fmt -check -recursive

Answer: D

Explanation:
-check Check if the input is formatted. Exit status will be 0 if all input is properly formatted and non-zero otherwise.
-recursive Also process files in subdirectories. By default, only the given directory (or current directory) is processed.

NEW QUESTION 254
- (Exam Topic 4)
Which of the following actions are performed during a terraform init?

A. Initializes downloaded and/or installed providers
B. Initializes the backend configuration
C. Provisions the declared resources in your configuration
D. Download the declared providers which are supported by HashiCorp

Answer: ABD

Explanation:
The terraform init command is used to initialize a working directory containing Terraform configuration files. This is the first command that should be run after writing a new Terraform configuration or cloning an existing one from version control. It is safe to run this command multiple times.
This command is always safe to run multiple times, to bring the working directory up to date with changes in the configuration. Though subsequent runs may give errors, this command will never delete your existing
configuration or state. terraform init command does * Copy a Source Module
* Backend Initialization
* Child Module Installation
* Plugin Installation https://www.terraform.io/docs/commands/init.html

NEW QUESTION 256
- (Exam Topic 4)
True or False? When using the Terraform provider for Vault, the tight integration between these HashiCorp tools provides the ability to mask secrets in the terraform plan and state files.

A. False
B. True

Answer: A

Explanation:

Currently, Terraform has no mechanism to redact or protect secrets that are returned via data sources, so secrets read via this provider will be persisted into the Terraform state, into any plan files, and in some cases in the console output produced while planning and applying. These artifacts must, therefore, all be protected accordingly.

NEW QUESTION 261
- (Exam Topic 4)
Your risk management organization requires that new AWS S3 buckets must be private and encrypted at rest. How can Terraform Enterprise automatically and proactively enforce this security control?

A. With a Sentinel policy, which runs before every apply
B. By adding variables to each TFE workspace to ensure these settings are always enabled
C. With an S3 module with proper settings for buckets
D. Auditing cloud storage buckets with a vulnerability scanning tool

Answer: A

Explanation:
https://docs.hashicorp.com/sentinel/intro/what
https://medium.com/hashicorp-engineering/enforcing-aws-s3-security-best-practice-using-terraform-sentinel-dd

NEW QUESTION 266
- (Exam Topic 4)
Open source Terraform can only import publicly-accessible and open-source modules.

A. True
B. False

Answer: B

Explanation:
Terraform can load modules from a public or private registry. This makes it possible to publish modules for others to use, and to use modules that others have published. Also, members of your organization might produce modules specifically crafted for your own infrastructure needs. Terraform Cloud and Terraform Enterprise both include a private module registry for sharing modules internally within your organization. Source: https://www.terraform.io/language/modules

NEW QUESTION 268
- (Exam Topic 4)
Terraform console provides an interactive command-line console for evaluating and experimenting with expressions. You can use it to test interpolations before using them in configurations and to interact with any values currently saved in state.
Which configuration consistency errors does terraform validate report?

A. A mix of spaces and tabs in configuration files
B. Differences between local and remote state
C. Terraform module isn't the latest version
D. Declaring a resource identifier more than once

Answer: D

Explanation:
validate will look for syntax errors "Declaring a resource identifier more than once" is a syntax error

NEW QUESTION 269
- (Exam Topic 4)
What are the benefits of using Infrastructure as Code? (select five)

A. Infrastructure as Code is relatively simple to learn and write, regardless of a user's prior experience with developing code
B. Infrastructure as Code provides configuration consistency and standardization among deployments
C. Infrastructure as Code is easily repeatable, allowing the user to reuse code to deploy similar, yet different resources
D. Infrastructure as Code gives the user the ability to recreate an application's infrastructure for disaster recovery scenarios
E. Infrastructure as Code easily replaces development languages such as Go and .Net for application development
F. Infrastructure as Code allows a user to turn a manual task into a simple, automated deployment (Correct)

Answer: ACDF

Explanation:

If you are new to infrastructure as code as a concept, it is the process of managing infrastructure in a file or files rather than manually configuring resources in a user interface.
A resource in this instance is any piece of infrastructure in a given environment, such as a virtual machine, security group, network interface, etc. At a high level, Terraform allows operators to use HCL to author files containing definitions of their desired resources on almost any provider (AWS, GCP, GitHub, Docker, etc) and automates the creation of those resources at the time of application.

NEW QUESTION 273
- (Exam Topic 4)
You have modified your Terraform configuration to fix a typo in the Terraform ID of a resource from aws_security_group.http to aws_security_group.http

Original configuration:

```
resource "aws_security_group" "htp" {
  name = "http"
  ingress {
    from_port  = "80"
    to_port    = "80"
    protocol   = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
}
```

Updated configuration:

```
resource "aws_security_group" "http" {
  name = "http"
  ingress {
    from_port  = "80"
    to_port    = "80"
    protocol   = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
}
```

Which of the following commands would you run to update the ID in state without destroying the resource?

A. terraform refresh
B. terraform apply
C. terraform mv aws-security-group.htp aws-security-group.http

Answer: C

Explanation:
The terraform state mv command changes which resource address in your configuration is associated with a particular real-world object. Use this to preserve an object when renaming a resource, or when moving a resource into or out of a child module.

NEW QUESTION 275
- (Exam Topic 4)
Which of the following is not a benefit of adopting infrastructure as code?

A. Automation
B. Versioning
C. Reusability of code
D. Interpolation

Answer: D

NEW QUESTION 277
- (Exam Topic 4)
Running terraform fmt without any flags in a directory with Terraform configuration files will check the formatting of those files without changing their contents.

A. True
B. False

Answer: B

Explanation:
The terraform fmt command is used to rewrite Terraform configuration files to a canonical format and style.

NEW QUESTION 281
- (Exam Topic 4)
True or False? terraform init cannot automatically download Community providers.

A. False
B. True

Answer: B

NEW QUESTION 282
- (Exam Topic 4)
What does this code do?

```
terraform {
    required_providers {
        aws = "~> 3.0"
    }
}
```

A. Requires any version of the AWS provider > = 3.0 and < 4.0
B. Requires any version of the AWS provider > = 3.0
C. Requires any version of the AWS provider after the 3.0 major release like 4.1
D. Requires any version of the AWS provider > 3.0

Answer: A

Explanation:
https://www.terraform.io/language/expressions/version-constraints#-3
Allows only the rightmost version component to increment. For example, to allow new patch releases within a specific minor release, use the full version number:
~> 1.0.4 will allow installation of 1.0.5 and 1.0.10 but not 1.1.0

NEW QUESTION 287
- (Exam Topic 4)
In the example below, where is the value of the DNS record's IP address originating from?
* 1. resource "aws_route53_record" "www"
* 2. {
* 3. zone_id = aws_route53_zone.primary.zone_id
* 4. name = "www.example.com"
* 5. type = "A"
* 6. ttl = "300"
* 7. records = [module.web_server.instance_ip_address] 8. }

A. The regular expression named module.web_server
B. The output of a module named web_server
C. By querying the AWS EC2 API to retrieve the IP address

D. Value of the web_server parameter from the variables.tf file

Answer: B

Explanation:
In a parent module, outputs of child modules are available in expressions as module.<MODULE NAME>.<OUTPUT NAME>.
For example, if a child module named web_server declared an output named instance_ip_address, you could access that value as module.web_server.instance_ip_address.

NEW QUESTION 289
- (Exam Topic 4)
How would you be able to reference an attribute from the vsphere_datacenter data source for use with the argument within the vsprere_folder resource in the following configuration?

```
data "vsphere_datacenter" "dc" {}

resource "vsphere_folder" "parent" {
            path = "Production"
            type = "vm"
    datacenter id = _____
}
```

A. vsphere_datacenter.dc.id
B. data.vsphere_datacenter.dc
C. data.dc,id
D. data.vsphere_datacenter.dc.id

Answer: D

NEW QUESTION 291
- (Exam Topic 4)
Terraform variable names are saved in the state file.

A. True
B. False

Answer: B

Explanation:
Terraform stores information about your infrastructure in a state file. This state file keeps track of resources created by your configuration and maps them to real-world resources. https://learn.hashicorp.com/tutorials/terraform/state-cli

NEW QUESTION 295
- (Exam Topic 4)
Which of the following is true about Terraform's implementation of infrastructure as code? (Choose two.)

A. It is only compatible with AWS infrastructure management
B. You cannot reuse infrastructure configuration
C. You can version your infrastructure configuration
D. It requires manual configuration of infrastructure resources
E. It allows you to automate infrastructure provisioning

Answer: CE

NEW QUESTION 300
- (Exam Topic 4)
From the code below, identify the implicit dependency:

A. The EIP with an id of ami-2757f631
B. The AMI used for the EC2 instance
C. The EC2 instance labeled web_server
D. The S3 bucket labeled company_data

Answer: C

NEW QUESTION 301
- (Exam Topic 4)
Multiple provider instances blocks for AWS can be part of a single configuration file?

A. False
B. True

Answer: B

Explanation:

You can optionally define multiple configurations for the same provider, and select which one to use on a per-resource or per-module basis. The primary reason for this is to support multiple regions for a cloud platform; other examples include targeting multiple Docker hosts, multiple Consul hosts, etc.

To include multiple configurations for a given provider, include multiple provider blocks with the same provider name, but set the alias meta-argument to an alias name to use for each additional configuration. For example:

The default provider configuration provider "aws" {

region = "us-east-1"

}

Additional provider configuration for west coast region provider "aws" {

alias = "west" region = "us-west-2"

}

The provider block without alias set is known as the default provider configuration. When alias is set, it creates an additional provider configuration. For providers that have no required configuration arguments, the implied empty configuration is considered to be the default provider configuration.

https://www.terraform.io/docs/configuration/providers.html#alias-multiple-provider-instances

NEW QUESTION 304

- (Exam Topic 4)

Which of the following locations can Terraform use as a private source for modules? (Choose two.)

A. Internally hosted SCM (Source Control Manager) platform
B. Public Terraform Module Registry
C. Private repository on GitHub
D. Public repository on GitHub

Answer: AC

NEW QUESTION 308

- (Exam Topic 4)

You are creating a Terraform configuration which needs to make use of multiple providers, one for AWS and one for Datadog.

Which of the following provider blocks would allow you to do this?

A)

B)

C)

```
terraform {
  provider "aws" {
    profile = var.aws_profile
    region  = var.aws_region
  }

  provider "datadog" {
    api_key = var.datadog_api_key
    app_key = var.datadog_app_key
  }
}
```

A. Option A
B. Option B
C. Option C

Answer: B

Explanation:
 https://www.terraform.io/language/providers/configuration

NEW QUESTION 312
- (Exam Topic 4)
When should Terraform configuration files be written when running terraform import on existing infrastructure?

A. Infrastructure can be imported without corresponding Terraform code
B. Terraform will generate the corresponding configuration files for you
C. You should write Terraform configuration files after the next terraform import is executed
D. Terraform configuration should be written before terraform import is executed

Answer: D

Explanation:
The current implementation of Terraform import can only import resources into the state. It does not generate configuration. A future version of Terraform will also generate configuration.
Because of this, prior to running terraform import it is necessary to write manually a resource configuration block for the resource, to which the imported object will be mapped.
Source: https://www.terraform.io/cli/import

NEW QUESTION 317
- (Exam Topic 4)
terraform destroy is the only way to remove infrastructure.

A. True
B. False

Answer: B

NEW QUESTION 321
- (Exam Topic 4)
Resources in terraform can have same identifiers(Resource type + Block name).

A. True
B. False

Answer: B

NEW QUESTION 322
- (Exam Topic 4)
True or False: Workspaces provide identical 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/workspaces/index.html https://www.terraform.io/docs/state/workspaces.html

NEW QUESTION 325
- (Exam Topic 4)
You decide to move a Terraform state file to Amazon S3 from another location. You write the code below into a file called\

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

You immediately run terraform apply but don't see any changes. Your state file didn't move. Which command
will migrate your current state file to the new S3 remote backend?

A. terraform push
B. terraform init
C. terraform refresh
D. terraform state

Answer: B

NEW QUESTION 327
- (Exam Topic 4)
Do terraform workspaces help in adding/allowing multiple state files for a single configuration?

A. True
B. False

Answer: A

NEW QUESTION 329
- (Exam Topic 4)
How would you reference the Volume IDs associated with the ebs_block_device blocks in this configuration?

```
resource "aws_instance" "example" {
  ami = "ami-abc123"
  instance_type - "t2.micro"

  ebs_block_device {
    device_name = "sda2"
    volume_size = 16
  }

  ebs_block_device {
    device_name = "sda3"
    volume_size = 20
  }
}
```

A. aws_instance.example.ebs_block_device.[*].volume_id
B. aws_instance.example.ebs_block_device.volume_id
C. aws_instance.example.ebs_block_device[sda2,sda3].volume_id
D. aws_instance.example.ebs_block_device.*.volume_id

Answer: A

Explanation:
https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/device_naming.html

NEW QUESTION 334
- (Exam Topic 4)
Your developers are facing a lot of problem while writing complex expressions involving difficult interpolations . They have to run the terraform plan every time and check whether there are errors , and also check terraform apply to print the value as a temporary output for debugging purposes. What should be done to avoid this?

A. Use terraform console command to have an interactive UI with full access to the underlying terraform state to run your interpolations , and debug at real-time.

B. Add a breakpoint in your code, using the watch keyword , and output the value to console for temporary debugging.
C. Use terraform zipmap function , it will be able to easily do the interpolations without complex code.
D. Use terraform console command to have an interactive UI , but you can only use it with local state , and it does not work with remote state.

Answer: A

Explanation:
The terraform console command provides an interactive console for evaluating expressions. This is useful for testing interpolations before using them in configurations, and for interacting with any values currently saved in state.
https://www.terraform.io/docs/commands/console.html

NEW QUESTION 339
- (Exam Topic 4)
Module version is required to reference a module on the Terraform Module Registry.

A. True
B. False

Answer: B

NEW QUESTION 344
- (Exam Topic 4)
You wanted to destroy some of the dependent resources from real infrastructure. You choose to delete those resources from your configuration file and run terraform plan and then apply. Which of the following way your resources would be destroyed?

A. Terraform can still determine the correct order for destruction from the state even when you delete one or more items from the configuration.
B. Those would be destroyed in the order in which they were written in the configuration file previously before you have deleted them from configuration file.
C. The resource will be destructed in random order as you have already deleted them from configuration.
D. You can not destroy resources by deleting them from configuration file and running plan and apply.

Answer: A

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.

NEW QUESTION 345
- (Exam Topic 4)
By default, where does Terraform store its state file?

A. Amazon S3 bucket
B. shared directory
C. remotely using Terraform Cloud
D. current working directory

Answer: D

Explanation:
By default, the state file is stored in a local file named "terraform.tfstate", but it can also be stored remotely, which works better in a team environment.

NEW QUESTION 350
- (Exam Topic 4)
A terraform apply can not _____ infrastructure.

A. import
B. provision
C. destroy
D. change

Answer: A

NEW QUESTION 355
- (Exam Topic 4)
How does Terraform determine dependencies between resources?

A. Terraform automatically builds a resource graph based on resources, provisioners, special meta-parameters, and the state file, if present.
B. Terraform requires all dependencies between resources to be specified using the depends_on parameter
C. Terraform requires resources in a configuration to be listed in the order they will be created to determine dependencies
D. Terraform requires resource dependencies to be defined as modules and sourced in order

Answer: A

Explanation:
https://learn.hashicorp.com/tutorials/terraform/dependencies

NEW QUESTION 360
- (Exam Topic 4)
You have modified your local Terraform configuration and ran terraform plan to review the changes. Simultaneously, your teammate manually modified the infrastructure component you are working on. Since you already ran terraform plan locally, the execution plan for terraform apply will be the same.

A. True
B. False

Answer: B

NEW QUESTION 362
- (Exam Topic 4)
Your organization has moved to AWS and has manually deployed infrastructure using the console. Recently, a decision has been made to standardize on Terraform for all deployments moving forward.
What can you do to ensure that all existing is managed by Terraform moving forward without interruption to existing services?

A. Submit a ticket to AWS and ask them to export the state of all existing resources and use terraform import to import them into the state file.
B. Delete the existing resources and recreate them using new a Terraform configuration so Terraform can manage them moving forward.
C. Resources that are manually deployed in the AWS console cannot be imported by Terraform.
D. Using terraform import, import the existing infrastructure into your Terraform state.

Answer: D

Explanation:
Terraform is able to import existing infrastructure. This allows us take resources we've created by some other means (i.e. via console) and bring it under Terraform management.
This is a great way to slowly transition infrastructure to Terraform.
The terraform import command is used to import existing infrastructure.
To import a resource, first write a resource block for it in our configuration, establishing the name by which it will be known to Terraform.
Example:
resource "aws_instance" "import_example" {
# ...instance configuration...
}
Now terraform import can be run to attach an existing instance to this resource configuration.
$ terraform import aws_instance.import_example i-03efafa258104165f aws_instance.import_example: Importing from ID "i-03efafa258104165f"...
aws_instance.import_example: Import complete!
Imported aws_instance (ID: i-03efafa258104165f) aws_instance.import_example: Refreshing state... (ID: i-03efafa258104165f) Import successful!
The resources that were imported are shown above. These resources are now in your Terraform state and will henceforth be managed by Terraform.
This command locates the AWS instance with ID i-03efafa258104165f (which has been created outside
Terraform) and attaches its existing settings, as described by the EC2 API, to the name aws_instance.import_example in the Terraform state.

NEW QUESTION 363
- (Exam Topic 4)
All Terraform Cloud tiers support team management and governance.

A. True
B. False

Answer: B

Explanation:
https://www.terraform.io/cloud-docs/overview
Terraform Cloud is a commercial SaaS product developed by HashiCorp. Many of its features are free for small teams, including remote state storage, remote runs, and VCS connections. We also offer paid plans for larger teams that include additional collaboration and governance features. Each higher paid upgrade plan is a strict superset of any lower plans — for example, the Team & Governance plan includes all of the features of the Team plan.

NEW QUESTION 365
- (Exam Topic 4)
terraform apply will fail if you have not run terraform plan first to update the plan output.

A. True
B. False

Answer: B

NEW QUESTION 366
- (Exam Topic 4)
In order to make a Terraform configuration file dynamic and/or reusable, static values should be converted to use what?

A. Input Parameters
B. Module
C. Regular Expressions
D. Output Value

Answer: A

Explanation:
Input variables serve as parameters for a Terraform module, allowing aspects of the module to be customized without altering the module's own source code, and

allowing modules to be shared between different configurations.
https://www.terraform.io/docs/configuration/variables.html

NEW QUESTION 371
- (Exam Topic 4)
In terraform, most resource dependencies are handled automatically. Which of the following statements describes best how terraform resource dependencies are handled?

A. Resource dependencies are identified and maintained in a file called resource.dependencie
B. Each terraform provider is required to maintain a list of all resource dependencies for the provider and it's included with the plugin during initialization when terraform init is execute
C. The file is located in the terraform.d folder.
D. The terraform binary contains a built-in reference map of all defined Terraform resource dependencies.Updates to this dependency map are reflected in terraform version
E. To ensure you are working with thelatest resource dependency map you much be running the latest version of Terraform.
F. Resource dependencies are handled automatically by the depends_on meta_argument, which is set to true by default.
G. Terraform analyses any expressions within a resource block to find references to other objects, and treats those references as implicit ordering requirements when creating, updating, or destroying resources.

Answer: D

Explanation:
https://www.terraform.io/docs/configuration/resources.html

NEW QUESTION 375
- (Exam Topic 4)
Terraform Cloud is more powerful when you integrate it with your version control system (VCS) provider. Select all the supported VCS providers from the answers below. (select four)

A. GitHub
B. CVS Version Control
C. Azure DevOps Server
D. Bitbucket Cloud
E. GitHub Enterprise

Answer: ACDE

Explanation:

Terraform Cloud supports the following VCS providers:
- https://www.terraform.io/docs/cloud/vcs/github.html
- https://www.terraform.io/docs/cloud/vcs/github.html
- https://www.terraform.io/docs/cloud/vcs/github-enterprise.html
- https://www.terraform.io/docs/cloud/vcs/gitlab-com.html
- https://www.terraform.io/docs/cloud/vcs/gitlab-eece.html
- https://www.terraform.io/docs/cloud/vcs/bitbucket-cloud.html
- https://www.terraform.io/docs/cloud/vcs/bitbucket-server.html
- https://www.terraform.io/docs/cloud/vcs/azure-devops-server.html
- https://www.terraform.io/docs/cloud/vcs/azure-devops-services.html https://www.terraform.io/docs/cloud/vcs/index.html#supported-vcs-providers

NEW QUESTION 376
- (Exam Topic 4)
Jack is a newbieto Terraform and wants to enable detailed logging to find all the details. Which environment variable does he need to set?

A. TF_help
B. TF LOG
C. TF_Debug
D. TF_var_log

Answer: B

NEW QUESTION 378
- (Exam Topic 4)
In order to reduce the time it takes to provision resources, Terraform uses parallelism. By default, how many resources will Terraform provision concurrently?

A. 5
B. 50
C. 10
D. 20

Answer: C

NEW QUESTION 383
- (Exam Topic 4)
Choose the answer that correctly completes the sentence: _____ backends support state locking.

A. All
B. No

C. Only local
D. Some

Answer: D

NEW QUESTION 388
- (Exam Topic 4)
While Terraform is generally written using the HashiCorp Configuration Language (HCL), what other syntax can Terraform are expressed in?

A. JSON
B. YAML
C. TypeScript
D. XML

Answer: A

Explanation:
The constructs in the Terraform language can also be expressed in JSON syntax, which is harder for humans to read and edit but easier to generate and parse programmatically.

NEW QUESTION 393
- (Exam Topic 4)
Which feature of Terraform allows multiple state files for a single configuration file depending upon the environment?

A. Terraform Modules
B. Terraform Enterprise
C. Terraform Workspaces
D. Terraform Remote Backends

Answer: C

NEW QUESTION 394
- (Exam Topic 4)
Named workspaces are not a suitable isolation mechanism for strong separation between staging and production?

A. True
B. False

Answer: A

Explanation:
Organizations commonly want to create a strong separation between multiple deployments of the same infrastructure serving different development stages (e.g. staging vs. production) or different internal teams. In this case, the backend used for each deployment often belongs to that deployment, with different credentials and access controls. Named workspaces are not a suitable isolation mechanism for this scenario.
https://www.terraform.io/docs/state/workspaces.html#when-to-use-multiple-workspaces

NEW QUESTION 399
- (Exam Topic 4)
Select all Operating Systems that Terraform is available for. (select five)

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

Answer: ABDEF

Explanation:

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

NEW QUESTION 400
- (Exam Topic 4)
If a DevOps team adopts AWS Cloud Formation as their standardized method for provisioning public cloud resources, which of the following scenarios poses a challenge for this team?

A. The team is asked to manage a new application stack built on AWS-natrve services
B. The organization decides to expand into Azure and wishes to deploy new infrastructure using their existing codebase
C. The team is asked to build a reusable code base that can deploy resources into any AWS region
D. The DevOps team is tasked with automating a manual provisioning process

Answer: B

NEW QUESTION 403

- (Exam Topic 4)
Using multi-cloud and provider-agnostic tools provides which of the following benefits?

A. Operations teams only need to learn and manage a single tool to manage infrastructure, regardless of where the infrastructure is deployed.
B. Increased risk due to all infrastructure relying on a single tool for management.
C. Can be used across major cloud providers and VM hypervisors.
D. Slower provisioning speed allows the operations team to catch mistakes before they are applied.

Answer: AC

Explanation:
Using a tool like Terraform can be advantageous for organizations deploying workloads across multiple public and private cloud environments. Operations teams only need to learn a single tool, single language, and can use the same tooling to enable a DevOps-like experience and workflows.

NEW QUESTION 407
- (Exam Topic 4)
You've used Terraform to deploy a virtual machine and a database. You want to replace this virtual machine instance with an identical one without affecting the database. What is the best way to achieve this using Terraform?

A. Use the Terraform taint command targeting the VMs then run Terraform plan and Terraform apply
B. Delete the Terraform VM resources from your Terraform code then run Terraform plan and terraform apply
C. Use the terraform apply command targeting the VM resources only
D. Use the terraform state rm command to remove the VM from state file

Answer: A

Explanation:
 https://www.terraform.io/cli/state/taint

NEW QUESTION 412
- (Exam Topic 4)
You cannot install third party plugins using terraform init.

A. True
B. False

Answer: B

Explanation:
https://www.terraform.io/cli/commands/init
For providers that are published in either the public Terraform Registry or in a third-party provider registry, terraform init will automatically find, download, and install the necessary provider plugins.

NEW QUESTION 414
- (Exam Topic 4)
After executing a terraform apply, you notice that a resource has a tilde (~) next to it. What does this infer?

A. The resource will be updated in place.
B. The resource will be created.
C. Terraform can't determine how to proceed due to a problem with the state file.
D. The resource will be destroyed and recreated.

Answer: A

Explanation:
The prefix -/+ means that Terraform will destroy and recreate the resource, rather than updating it in-place. The prefix ~ means that some attributes and resources can be updated in-place.
$ terraform apply
aws_instance.example: Refreshing state... [id=i-0bbf06244e44211d1] An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
-/+ destroy and then create replacement Terraform will perform the following actions:
# aws_instance.example must be replaced
-/+ resource "aws_instance" "example" {
~ ami = "ami-2757f631" -> "ami-b374d5a5" # forces replacement
~ arn = "arn:aws:ec2:us-east-1:130490850807:instance/i-0bbf06244e44211d1" -> (known after apply)
~ associate_public_ip_address = true -> (known after apply)
~ availability_zone = "us-east-1c" -> (known after apply)
~ cpu_core_count = 1 -> (known after apply)
~ cpu_threads_per_core = 1 -> (known after apply)
- disable_api_termination = false -> null
- ebs_optimized = false -> null get_password_data = false
+ host_id = (known after apply)
~ id = "i-0bbf06244e44211d1" -> (known after apply)
~ instance_state = "running" -> (known after apply) instance_type = "t2.micro"
~ ipv6_address_count = 0 -> (known after apply)
~ ipv6_addresses = [] -> (known after apply)
+ key_name = (known after apply)
- monitoring = false -> null
+ network_interface_id = (known after apply)
+ password_data = (known after apply)

```
+ placement_group = (known after apply)
~ primary_network_interface_id = "eni-0f1ce5bdae258b015" -> (known after apply)
~ private_dns = "ip-172-31-61-141.ec2.internal" -> (known after apply)
~ private_ip = "172.31.61.141" -> (known after apply)
~ public_dns = "ec2-54-166-19-244.compute-1.amazonaws.com" -> (known after apply)
~ public_ip = "54.166.19.244" -> (known after apply)
~ security_groups = [
- "default",
] -> (known after apply) source_dest_check = true
~ subnet_id = "subnet-1facdf35" -> (known after apply)
~ tenancy = "default" -> (known after apply)
~ volume_tags = {} -> (known after apply)
~ vpc_security_group_ids = [
- "sg-5255f429",
] -> (known after apply)
- credit_specification {
- cpu_credits = "standard" -> null
}
+ ebs_block_device {
+ delete_on_termination = (known after apply)
+ device_name = (known after apply)
+ encrypted = (known after apply)
+ iops = (known after apply)
+ snapshot_id = (known after apply)
+ volume_id = (known after apply)
+ volume_size = (known after apply)
+ volume_type = (known after apply)
}
+ ephemeral_block_device {
+ device_name = (known after apply)
+ no_device = (known after apply)
+ virtual_name = (known after apply)
}
+ network_interface {
+ delete_on_termination = (known after apply)
+ device_index = (known after apply)
+ network_interface_id = (known after apply)
}
~ root_block_device {
~ delete_on_termination = true -> (known after apply)
~ iops = 100 -> (known after apply)
~ volume_id = "vol-0079e485d9e28a8e5" -> (known after apply)
~ volume_size = 8 -> (known after apply)
~ volume_type = "gp2" -> (known after apply)
}
}
Plan: 1 to add, 0 to change, 1 to destroy.
```

NEW QUESTION 417
- (Exam Topic 4)
When using parent/child modules to deploy infrastructure, how would you export a value from one module to import into another module.
For example, a module dynamically deploys an application instance or virtual machine, and you need the IP address in another module to configure a related DNS record in order to reach the newly deployed application.

A. Export the value using terraform export and input the value using terraform input.
B. Configure the pertinent provider's configuration with a list of possible IP addresses to use.
C. Configure an output value in the application module in order to use that value for the DNS module.
D. Preconfigure the IP address as a parameter in the DNS module.

Answer: C

Explanation:
Output values are like the return values of a Terraform module, and have several uses:
* A child module can use outputs to expose a subset of its resource attributes to a parent module.
* A root module can use outputs to print certain values in the CLI output after running terraform apply.
* When using remote state, root module outputs can be accessed by other configurations via a terraform_remote_state data source.
https://www.terraform.io/docs/configuration/outputs.html

NEW QUESTION 418
- (Exam Topic 4)
How do you specify a module's version when publishing it to the public Terraform Module Registry?

A. The module's configuration page on the Terraform Module Registry
B. Terraform Module Registry does not support versioning modules
C. The release tags in the associated repo Most Voted
D. The module's Terraform code

Answer: C

Explanation:
https://www.terraform.io/registry/modules/publish

NEW QUESTION 423
- (Exam Topic 4)
What does terraform refresh modify?

A. Your cloud infrastructure
B. Your Terraform plan
C. Your state file
D. Your Terraform configuration

Answer: C

NEW QUESTION 427
- (Exam Topic 4)
You have a simple Terraform configuration containing one virtual machine (VM) in a cloud provider. You run terraform apply and the VM is created successfully.
What will happen if you terraform apply again immediately afterwards without changing any Terraform code?

A. Terraform will terminate and recreate the VM
B. Terraform will create another duplicate VM
C. Terraform will apply the VM to the state file
D. Nothing

Answer: D

NEW QUESTION 432
- (Exam Topic 4)
resource "aws_s3_bucket" "example" { bucket = "my-test-s3-terraform-bucket" ...} resource "aws_iam_role" "test_role" { name = "test_role" ...}
Due to the way that the application code is written , the s3 bucket must be created before the test role is created , otherwise there will be a problem. How can you ensure that?

A. This will already be taken care of by terraform native implicit dependenc
B. Nothing else needs to be done from your end.
C. Add explicit dependency using depends_on . This will ensure the correct order of resource creation.
D. Create 2 separate terraform config scripts , and run them one by one , 1 for s3 bucket , and another for IAM role , run the S3 bucket script first.
E. This is not possible to control in terraform . Terraform will take care of it in a native way , and create a dependency graph that is best suited for the parallel resource creation.

Answer: B

Explanation:
Use the depends_on meta-argument to handle hidden resource dependencies that Terraform can't automatically infer.
Explicitly specifying a dependency is only necessary when a resource relies on some other resource's behavior but doesn't access any of that resource's data in its arguments.

NEW QUESTION 434
- (Exam Topic 4)
While attempting to deploy resources into your cloud provider using Terraform. you begin to see some odd behavior and experience sluggish responses. In order to troubleshoot you decide to turn on Terraform debugging. Which environment variables must be configured to make Terraform's logging more verbose?

A. TF_10G_PATM
B. TF_LOG
C. TF_10G_LEVEL
D. TF.LOG.FUE

Answer: B

Explanation:
https://www.terraform.io/internals/debugging

NEW QUESTION 437
- (Exam Topic 4)
When do you need to explicitly execute terraform refresh?

A. Before every terraform plan
B. Before every terraform apply
C. Before every terraform import
D. None of the above

Answer: D

Explanation:
Wherever possible, avoid using terraform refresh explicitly and instead rely on Terraform's behavior of automatically refreshing existing objects as part of creating a normal plan. Source: https://www.terraform.io/cli/commands/refresh

NEW QUESTION 441
- (Exam Topic 4)
You need to specify a dependency manually. What resource meta-parameter can you use lo make sure Terraform respects thee dependency?

Type your answer in the field provided. The text field is not case-sensitive and all variations of the correct answer are accepted.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
depends_on

NEW QUESTION 443
- (Exam Topic 4)
Which of the following is true about terraform apply? (Choose two.)

A. It only operates on infrastructure defined in the current working directory or workspace
B. You must pass the output of a terraform plan command to it
C. Depending on provider specification, Terraform may need to destroy and recreate your infrastructure resources
D. By default, it does not refresh your state file to reflect current infrastructure configuration
E. You cannot target specific resources for the operation

Answer: AC

Explanation:
https://www.terraform.io/cli/run

NEW QUESTION 445
- (Exam Topic 4)
Once a new Terraform backend is configured with a Terraform code block, which command(s) is (are) used to migrate the state file?

A. terraform apply
B. terraform push
C. terraform destroy, then terraform apply
D. terraform init

Answer: B

Explanation:
https://www.terraform.io/cli/commands/state/push

NEW QUESTION 450
- (Exam Topic 4)
You need to write some Terraform code that adds 42 firewall rules to a security group as shown in the example.

```
resource "aws_security_group" "many_rules" {
  name = "many-rules"
  ingress {
    from_port = 443
    to_port = 443
    protocol = "tcp"
    cidr_blocks = "0.0.0.0/0"
  }
}
```

What can you use to avoid writing 42 different nested ingress config blocks by hand?

A. A count loop
B. A for block
C. A for each block
D. A dynamic block

Answer: D

Explanation:
A dynamic block acts much like a for expression, but produces nested blocks instead of a complex typed value. It iterates over a given complex value, and generates a nested block for each element of that complex value. Reference: https://www.terraform.io/language/expressions/dynamic-blocks

NEW QUESTION 455
- (Exam Topic 4)
Which type of block fetches or computes information for use elsewhere in a Terraform configuration?

A. provider
B. resource
C. local

D. data

Answer: D

Explanation:
Data sources allow data to be fetched or computed for use elsewhere in Terraform configuration. Use of data sources allows a Terraform configuration to build on information defined outside of Terraform, or defined by another separate Terraform configuration.

NEW QUESTION 456
- (Exam Topic 4)
Which of the following is the safest way to inject sensitive values into a Terraform Cloud workspace?

A. Write the value to a file and specify the file with the -var-file flag
B. Set a value for the variable in the UI and check the "Sensitive" check box
C. Edit the state file directly just before running terraform apply
D. Set the variable value on the command line with the -var flag

Answer: B

Explanation:
-var and -var-file overwrite workspace-specific and variable set variables that have the same key. From the workspace, variable can be added and checked off as being sensitive. Reference: https://www.terraform.io/cloud-docs/workspaces/variables/managing-variables#loading-variables-from-files
https://www.terraform.io/cloud-docs/workspaces/variables

NEW QUESTION 459
- (Exam Topic 4)
terraform validate reports HCL syntax errors.

A. True
B. False

Answer: A

NEW QUESTION 460
- (Exam Topic 4)
Which of the following does terraform apply change after you approve the execution plan? Choose two correct answers.

A. The execution plan
B. Terraform code
C. Cloud infrastructure
D. State file
E. The .terraform directory

Answer: CD

NEW QUESTION 464
- (Exam Topic 4)
What does state locking accomplish?

A. Copies the state file from memory to disk
B. Encrypts any credentials stored within the state file
C. Blocks Terraform commands from modifying the state file
D. Prevents accidental deletion of the state file

Answer: C

Explanation:
If supported by your backend, Terraform will lock your state for all operations that could write state. This prevents others from acquiring the lock and potentially corrupting your state. Source: https://www.terraform.io/language/state/locking

NEW QUESTION 467
- (Exam Topic 4)
A junior admin accidentally deleted some of your cloud instances. What does Terraform do when you run terraform apply?

A. Build a completely brand new set of infrastructure
B. Tear down the entire workspace infrastructure and rebuild it
C. Rebuild only the instances that were deleted Most Voted
D. Stop and generate an error message about the missing instances

Answer: C

NEW QUESTION 471
- (Exam Topic 4)
A variable az has the following default value. What will be the datatype of the variable? az=["us-west-1a","us-east-1a"]

A. Object

B. List
C. Map
D. String

Answer: B

NEW QUESTION 472
- (Exam Topic 4)
What is the best and easiest way for Terraform to read and write secrets from HashiCorp Vault?

A. Vault provider
B. API access using the AppRole auth method
C. integration with a tool like Jenkins
D. CLI access from the same machine running Terraform

Answer: A

NEW QUESTION 473
- (Exam Topic 4)
Which is the best way to specify a tag of v1.0.0 when referencing a module stored in Git (for example git::https://example.com/vpc.git)?

A. Append ref=v1. 0. 0 argument to the source path Most Voted
B. Add version = "1.0.0" parameter to module block
C. Nothing " modules stored on GitHub always default to version 1.0.0
D. Modules stored on GitHub do not support versioning

Answer: A

Explanation:
https://www.terraform.io/language/modules/sources#selecting-a-revision

NEW QUESTION 474
- (Exam Topic 4)
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.

NEW QUESTION 478
- (Exam Topic 4)
What does Terraform use providers for? (Choose three.)

A. Provision resources for on-premises infrastructure services
B. Simplify API interactions
C. Provision resources for public cloud infrastructure services
D. Enforce security and compliance policies
E. Group a collection of Terraform configuration files that map to a single state file

Answer: ABC

NEW QUESTION 481
- (Exam Topic 4)
What are some of the features of Terraform state? (select three)

A. inspection of cloud resources
B. determining the correct order to destroy resources
C. mapping configuration to real-world resources
D. increased performance

Answer: CD

NEW QUESTION 486
- (Exam Topic 4)
Choose the best option from below to make Terraform code more user configuration-centric.

A. Variables
B. Local values

C. Input Variable
D. Modules

Answer: C

NEW QUESTION 487
- (Exam Topic 4)
Which of the following commands will launch the Interactive console for Terraform interpolations?

A. terraform console
B. terraform cli
C. terraform
D. terraform cmdline

Answer: B

Explanation:
https://www.terraform.io/docs/commands/console.html

NEW QUESTION 488
- (Exam Topic 4)
True or False? By default, Terraform destroy will prompt for confirmation before proceeding.

A. False
B. True

Answer: B

NEW QUESTION 490
- (Exam Topic 4)
As a member of the operations team, you need to run a script on a virtual machine created by Terraform. Which provisioned is best to use in your Terraform code?

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

Answer: D

Explanation:
 https://www.terraform.io/language/resources/provisioners/remote-exec

NEW QUESTION 491
- (Exam Topic 4)
You have just developed a new Terraform configuration for two virtual machines with a cloud provider. You would like to create the infrastructure for the first time. Which Terraform command should you run first?

A. terraform apply
B. terraform plan
C. terraform show
D. terraform init

Answer: D

NEW QUESTION 494
- (Exam Topic 4)
Which of the below backends support state locking?

A. S3
B. consul
C. azurerm
D. artifactory

Answer: ABC

NEW QUESTION 496
- (Exam Topic 4)
As a member of an operations team that uses infrastructure as code (IaC) practices, you are tasked with making a change to an infrastructure stack running in a public cloud. Which pattern would follow IaC best practices for making a change?

A. Make the change via the public cloud API endpoint
B. Make the change programmatically via the public cloud CLI
C. Submit a pull request and wait for an approved merge of the proposed changes
D. Use the public cloud console to make the change after a database record has been approved
E. Clone the repository containing your infrastructure code and then run the code

Answer: C

NEW QUESTION 497
- (Exam Topic 4)
Which of the following is not a way to trigger terraform destroy ?

A. Passing ---destroy at the end of apian request
B. Running terraform destroy from the correct directory and then typing "yes" when prompted in the CLI
C. Using the destroy command with auto approve
D. Delete the state file and run terraform apply

Answer: A

NEW QUESTION 500
- (Exam Topic 4)
Given the Terraform configuration below, in which order will the resources be created?

A. Larger image
B. resources will be created simultaneously
C. aws_eip will be created first aws_instance will be created second
D. aws_instance will be created first aws_eip will be created second

Answer: D

Explanation:
The aws_instance will be created first, and then aws_eip will be created second due to the aws_eip's resource dependency of the aws_instance id

NEW QUESTION 502
- (Exam Topic 4)
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: AE

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

NEW QUESTION 505
- (Exam Topic 4)
What does terraform destroy do?

A. Destroy all infrastructure in the Terraform state file
B. Destroy all Terraform code files in the current directory while leaving the state file intact
C. Destroy all infrastructure in the configured Terraform provider
D. Destroy the Terraform state file while leaving infrastructure intact

Answer: A

Explanation:
The terraform destroy command terminates resources managed by your Terraform project. This command is the inverse of terraform apply in that it terminates all the resources specified in your Terraform state. It does not destroy resources running elsewhere that are not managed by the current Terraform project.
https://learn.hashicorp.com/tutorials/terraform/aws-destroy

NEW QUESTION 509
- (Exam Topic 4)
You have been working in a Cloud provider account that is shared with other team members. You previously used Terraform to create a load balancer that is listening on port 80. After some application changes, you updated the Terraform code to change the port to 443.
You run terraform plan and see that the execution plan shows the port changing from 80 to 443 like you intended, and step away to grab some coffee.
In the meantime, another team member manually changes the load balancer port to 443 through the Cloud provider console before you get back to your desk.
What will happen when you terraform apply upon returning to your desk?

A. Terraform will not make any changes to the Load Balancer and will update the state file to reflect any changes made.
B. Terraform will change the port back to 80 in your code
C. Terraform will change the load balancer port to 80, and) then change it back to 443
D. Terraform will fail with in error because the state file is no longer accurate

Answer: A

NEW QUESTION 512
......