Exam Questions TA-002-P

HashiCorp Certified: Terraform Associate

https://www.2passeasy.com/dumps/TA-002-P/

NEW QUESTION 1
- (Exam Topic 1)
What command should you run to display all workspaces for the current configuration?

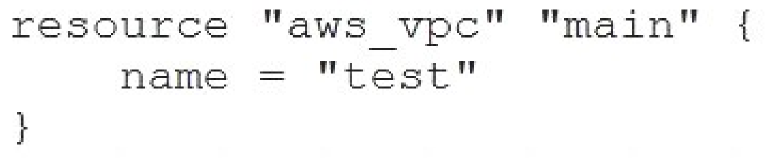
A. terraform workspace
B. terraform workspace show
C. terraform workspace list
D. terraform show workspace

Answer: C

Explanation:

terraform workspace list
The command will list all existing workspaces.
Reference: https://www.terraform.io/docs/cli/commands/workspace/list.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)
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 delete the VM using the cloud provider console, and run terraform apply again without changing any Terraform code?

A. Terraform will remove the VM from state file
B. Terraform will report an error
C. Terraform will not make any changes
D. Terraform will recreate the VM

Answer: D

NEW QUESTION 4
- (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 5
- (Exam Topic 1)
When should you use the force-unlock command?

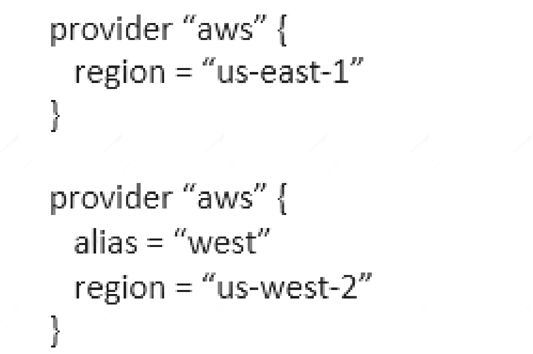
A. You see a status message that you cannot acquire the lock
B. You have a high priority change
C. Automatic unlocking failed
D. Your apply failed due to a state lock

Answer: C

Explanation:

Be very careful with this command. If you unlock the state when someone else is holding the lock it could cause multiple writers. Force unlock should only be used to unlock your own lock in the situation where automatic unlocking failed. Source: https://www.terraform.io/language/state/locking
https://www.terraform.io/cli/commands/force-unlock

NEW QUESTION 6
- (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?

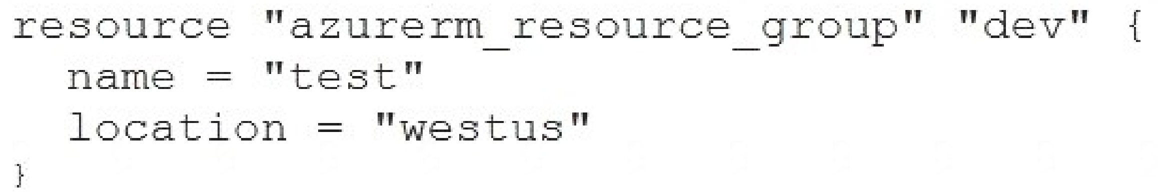
A. alias = west
B. provider = west
C. provider = aws.west
D. alias = aws.west

Answer: C

Explanation:

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

NEW QUESTION 7
- (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 8
- (Exam Topic 1)
You have used Terraform to create an ephemeral development environment in the cloud and are now ready to destroy all the infrastructure described by your Terraform configuration. To be safe, you would like to first see all the infrastructure that will be deleted by Terraform.
Which command should you use to show all of the resources that will be deleted? (Choose two.)

A. Run terraform plan -destroy.
B. This is not possibl
C. You can only show resources that will be created.
D. Run terraform state rm *.
E. Run terraform destroy and it will first output all the resources that will be deleted before prompting for approval.

Answer: AD

Explanation:

Reference: https://www.terraform.io/docs/cli/commands/state/rm.html

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

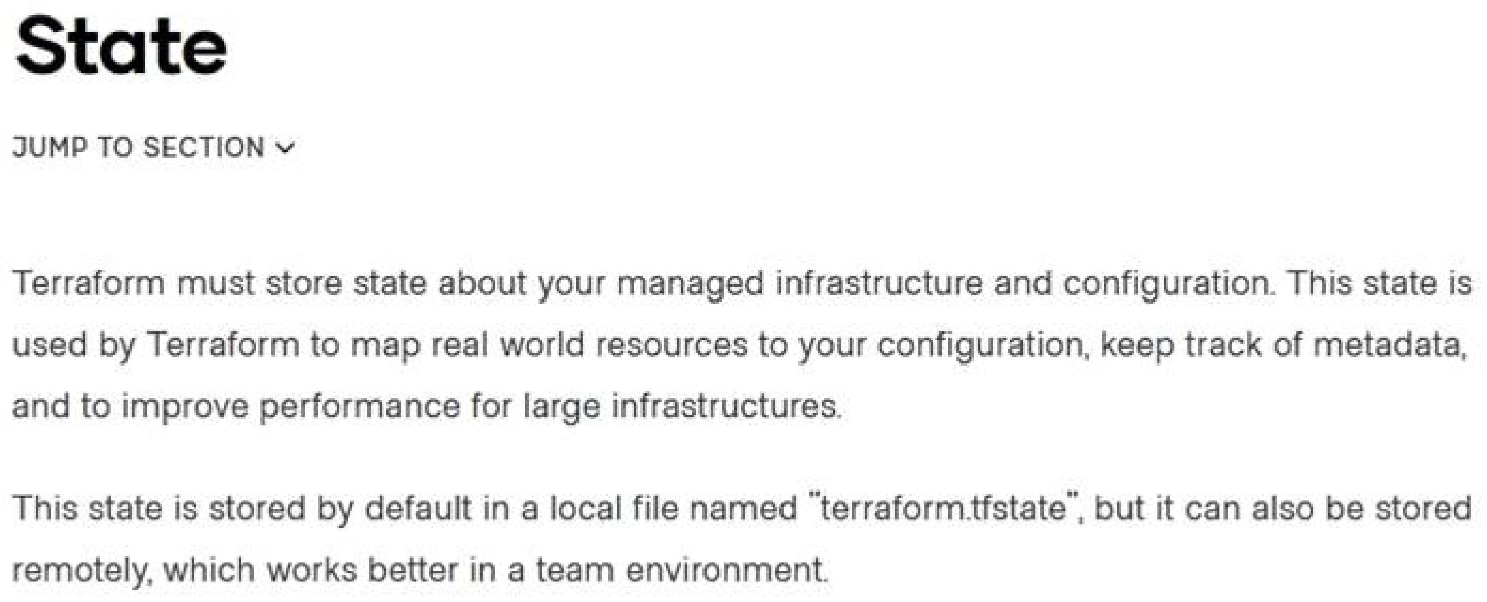
Explanation:
The terraform taint command manually marks a Terraform-managed resource as tainted, forcing it to be destroyed and recreated on the next apply.
Reference: https://www.terraform.io/docs/cli/commands/taint.html

NEW QUESTION 10
- (Exam Topic 1)
FILL BLANK
What is the name of the default file where Terraform stores the state?
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:
"This state is stored by default in a local file named "terraform.tfstate", but it can also be stored remotely, which works better in a team environment."
https://www.terraform.io/language/state



State

JUMP TO SECTION

Terraform must store state about your managed infrastructure and configuration. This state is used by Terraform to map real world resources to your configuration, keep track of metadata, and to improve performance for large infrastructures.

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

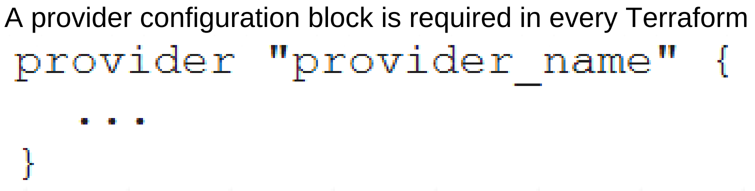
NEW QUESTION 10
- (Exam Topic 1)
Which two steps are required to provision new infrastructure in the Terraform workflow? (Choose two.)

A. Destroy
B. Apply
C. Import
D. Init
E. Validate

Answer: BD

Explanation:
Reference: https://www.terraform.io/guides/core-workflow.html

NEW QUESTION 13
- (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 17
- (Exam Topic 1)
A terraform apply can not ______ infrastructure.

A. change
B. destroy

C. provision
D. import

Answer: D

Explanation:
https://www.educative.io/answers/what-is-the-command-to-destroy-infrastructure-in-terraform

NEW QUESTION 21
- (Exam Topic 1)
Terraform can run on Windows or Linux, but it requires a Server version of the Windows operating system.

A. True
B. False

Answer: B

Explanation:
https://www.terraform.io/downloads

NEW QUESTION 25
- (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 28
- (Exam Topic 1)
Only the user that generated a plan may apply it.

A. True
B. False

Answer: B

NEW QUESTION 33
- (Exam Topic 1)
You have declared a variable called var.list which is a list of objects that all have an attribute id. Which options will produce a list of the IDs? (Choose two.)

A. { for o in var.list : o => o.id }
B. var.list[*].id
C. [var.list[*].id]
D. [for o in var.list : o.id]

Answer: BD

Explanation:
https://www.terraform.io/language/expressions/splat
A splat expression provides a more concise way to express a common operation that could otherwise be performed with a for expression.

NEW QUESTION 34
- (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 36
- (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 38
- (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 43
- (Exam Topic 1)
You have never used Terraform before and would like to test it out using a shared team account for a cloud provider. The shared team account already contains 15 virtual machines (VM). You develop a Terraform configuration containing one VM, perform terraform apply, and see that your VM was created successfully.
What should you do to delete the newly-created VM with Terraform?

A. The Terraform state file contains all 16 VMs in the team accoun
B. Execute terraform destroy and select the newly-created VM.
C. The Terraform state file only contains the one new V
D. Execute terraform destroy.
E. Delete the Terraform state file and execute Terraform apply.
F. Delete the VM using the cloud provider console and terraform apply to apply the changes to the Terraform state file.

Answer: B

Explanation:
You develop a Terraform configuration containing one VM, perform terraform apply, and see that your VM was created successfully. read the question carefully "Terraform configuration containing one VM, perform terraform apply" so only one VM is in state file.

NEW QUESTION 48
- (Exam Topic 1)
What is the workflow for deploying new infrastructure with Terraform?

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

Answer: D

Explanation:
Reference: https://www.google.com/search?q=Write+a+Terraform+configuration%2C+run+terraform+init%2C+run+terraform+plan+to+view+planned+infrastructure+changes%2C+and+terraform+apply+to+create+new+infrastructure.&oq=Write+a+Terraform+configuration%2C+run+terraform+init%2C+run+terraform+plan+to+view+planned+infrastructure+changes%2C+and+terraform+apply+to+create+new+infrastructure.&aqs=chrome..69i57.556j0j7&sourceid=chrome&ie=UTF-8

NEW QUESTION 52
- (Exam Topic 1)
If a module declares a variable with a default, that variable must also be defined within the module.

A. True
B. False

Answer: B

NEW QUESTION 56
- (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 57
- (Exam Topic 1)
HashiCorp Configuration Language (HCL) supports user-defined functions.

A. True
B. False

Answer: B

Explanation:
https://www.terraform.io/language/functions
The Terraform language does not support user-defined functions, and so only the functions built into the language are available for use

NEW QUESTION 62
- (Exam Topic 1)
In Terraform 0.13 and above, outside of the required_providers block, Terraform configurations always refer to providers by their local names.

A. True
B. False

Answer: A

Explanation:
Outside of the required_providers block, Terraform configurations always refer to providers by their local names.
Reference: https://www.terraform.io/docs/language/providers/requirements.html https://www.terraform.io/language/providers/requirements#local-names

NEW QUESTION 63
- (Exam Topic 1)
Terraform and Terraform providers must use the same major version number in a single configuration.

A. True
B. False

Answer: B

Explanation:
https://www.terraform.io/language/expressions/version-constraints#terraform-core-and-provider-versions

NEW QUESTION 68
- (Exam Topic 1)
All standard backend types support state storage, locking, and remote operations like plan. apply and destroy.

A. True
B. False

Answer: B

Explanation:
https://www.terraform.io/language/settings/backends/configuration
"Some of these backends act like plain remote disks for state files, while others support locking the state while operations are being performed. This helps prevent conflicts and inconsistencies. The built-in backends listed are the only backends. You cannot load additional backends as plugins."

NEW QUESTION 72
- (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 75
- (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 77
- (Exam Topic 1)
How is the Terraform remote backend different than other state backends such as S3, Consul, etc.?

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

Answer: A

Explanation:
Backends define where Terraform's state snapshots are stored. A given Terraform configuration can either specify a backend, integrate with Terraform Cloud, or do neither and default to storing state locally.
If you and your team are using Terraform to manage meaningful infrastructure, we recommend using the remote backend with Terraform Cloud or Terraform Enterprise.
Reference: https://www.terraform.io/docs/language/settings/backends/index.html

NEW QUESTION 81
- (Exam Topic 1)
How is terraform import run?

A. As a part of terraform init
B. As a part of terraform plan
C. As a part of terraform refresh
D. By an explicit call
E. All of the above

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. While this may seem tedious, it still gives Terraform users an avenue for importing existing resources."
https://www.terraform.io/cli/import/usage

NEW QUESTION 85
- (Exam Topic 1)
You have multiple team members collaborating on infrastructure as code (IaC) using Terraform, and want to apply formatting standards for readability.
How can you format Terraform HCL (HashiCorp Configuration Language) code according to standard Terraform style convention?

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

Answer: A

Explanation:
https://www.terraform.io/cli/commands/fmt

NEW QUESTION 90
- (Exam Topic 2)
Which one of the following command will rewrite Terraform configuration files to a canonical format and style.

A. terraform graph -h
B. terraform init
C. terraform graph
D. terraform fmt

Answer: D

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 Terrra Terraform language style conventions, along with other minor adjustments for readability.

NEW QUESTION 91
- (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 95
- (Exam Topic 2)
Which of the following clouds does not have a provider maintained HashiCorp?

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

Answer: A

Explanation:
IBM Cloud does not have a provider maintained by HashiCorp, although IBM Cloud does maintain their own Terraform provider.
https://www.terraform.io/docs/providers/index.html

NEW QUESTION 98
- (Exam Topic 2)
The current implementation of Terraform import can only import resources into the state. It does not generate configuration.

A. False
B. True

Answer: B

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.
While this may seem tedious, it still gives Terraform users an avenue for importing existing resources. https://www.terraform.io/docs/import/index.html#currently-state-only

NEW QUESTION 103
- (Exam Topic 2)
Terraform import command can import resources into modules as well directly into the root of your state.

A. True
B. False

Answer: A

Explanation:
Import will find the existing resource from ID and import it into your Terraform state at the given ADDRESS. ADDRESS must be a valid resource address. Because any resource address is valid, the import command can import resources into modules as well directly into the root of your state.
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. For 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.
As a result of the above command, the resource is recorded in the state file. We can now run terraform plan to see how the configuration compares to the imported resource, and make any adjustments to the configuration to align with the current (or desired) state of the imported object.
https://www.terraform.io/docs/commands/import.html

NEW QUESTION 107
- (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 112
- (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 113
- (Exam Topic 2)
What allows you to conveniently switch between multiple instances of a single configuration within its single backend?

A. Local backends
B. Providers
C. Remote backends
D. Workspaces

Answer: D

Explanation:
Named workspaces allow conveniently switching between multiple instances of a single configuration within its single backend. ... A common use for multiple workspaces is to create a parallel, distinct copy of a set of infrastructure in order to test a set of changes before modifying the main production infrastructure.
Workspaces, allowing multiple states to be associated with a single configuration. The configuration still has only one backend, but multiple distinct instances of that configuration to be deployed without configuring a new backend or changing authentication credentials.
https://www.terraform.io/docs/state/workspaces.html

NEW QUESTION 116
- (Exam Topic 2)
You are using a terraform operation that writes state. Unfortunately automatic state unlocking has failed for that operation. Which of the below commands can be used to remove the already acquired lock on the state?

A. terraform unlock
B. terraform force-unlock
C. terraform state unlock
D. None of the above

Answer: B

Explanation:
Command: force-unlock
Manually unlock the state for the defined configuration.
This will not modify your infrastructure. This command removes the lock on the state for the current configuration. The behavior of this lock is dependent on the backend being used. Local state files cannot be unlocked by another process.
https://www.terraform.io/docs/commands/force-unlock.html https://www.terraform.io/docs/state/locking.html
Terraform has a force-unlock command to manually unlock the state if unlocking failed.
If you unlock the state when someone else is holding the lock it could cause multiple writers. Force unlock should only be used to unlock your own lock in the situation where automatic unlocking failed.

NEW QUESTION 121
- (Exam Topic 2)
How does Terraform handle working with so many providers?

A. Terraform ships with all of the plugins embedded in the Terraform binary.
B. Terraform uses a plugin architecture for providers and only installs the provider plugins required by your configuration in the configuration's working directory.

C. Terraform uses a plugin architecture for providers and only installs the provider plugins required by your configuration in a shared, system-wide plugins directory.
D. Terraform allows you to select the providers you want to support during the Terraform installation process.

Answer: B

Explanation:
Terraform is built on a plugin-based architecture. All providers and provisioners that are used in Terraform configurations are plugins, even the core types such as AWS and Heroku. Users of Terraform are able to write new plugins in order to support new functionality in Terraform.

NEW QUESTION 124
- (Exam Topic 2)
Please identify the offerings which are unique to Terraform Enterprise, and not available in either Terraform OSS, or Terraform Cloud. Select four.

A. Audit Logs
B. Private Network Connectivity
C. VCS Integration
D. Sentinel
E. Clustering

Answer: ABE

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

NEW QUESTION 127
- (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 129
- (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 132
- (Exam Topic 3)
Your manager has instructed you to start using terraform for the entire infra provisioning of the application stack. There are 4 environments – DEV , QA , UAT , and PROD. The application team has asked for complete segregation between these environments including the backend , state , and also configurations ,since there will be unique resources in different environments . What is the possible way to structure the terraform code to facilitate that.

A. Completely separate the working directories , keep one for each environment . For each working directory , maintain a separate configuration file , variables file , and map to a different backend.
B. Completely separate the working directories , keep one for each environment . For each working directory , maintain a separate configuration file , variables file , and map to the same backend.
C. Implement terraform workspaces , and map each environment with one workspace.
D. Enable remote backend storage . Configure 4 different backend storages , one for each environment.

Answer: A

Explanation:
In particular, 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

NEW QUESTION 135
- (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 137
- (Exam Topic 3)
Which of the following is the right substitute for static values that can make Terraform configuration file more dynamic and reusable?

A. Output value
B. Input parameters
C. Functions
D. Modules

Answer: B

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.

NEW QUESTION 141
- (Exam Topic 3)
Which of the below options is a valid interpolation syntax for retrieving a data source?

A. ${google_storage_bucket.backend}
B. ${azurerm_resource_group.test.data}
C. ${aws_instance.web.id.data}
D. ${data.google_dns_keys.foo_dns_keys.key_signing_keys[0].ds_record}

Answer: D

Explanation:
Data source attributes are interpolated with the general syntax data.TYPE.NAME.ATTRIBUTE. The interpolation for a resource is the same but without the data. prefix (TYPE.NAME.ATTRIBUTE).
https://www.terraform.io/docs/configuration-0-11/interpolation.html#attributes-of-a-data-source

NEW QUESTION 143
- (Exam Topic 3)
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. What command will do this?

A. terraform taint
B. terraform apply
C. terraform graph
D. terraform refresh

Answer: A

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.
This example will taint a single resource:
$ terraform taint aws_security_group.allow_all
The resource aws_security_group.allow_all in the module root has been marked as tainted. https://www.terraform.io/docs/commands/taint.html

NEW QUESTION 145
- (Exam Topic 3)
Your company has been using Terraform Cloud for a some time now . But every team is creating their own modules , and there is no standardization of the modules , with each team creating the resources in their own unique way . You want to enforce a standardization of the modules across the enterprise . What should be your approach.

A. Create individual workspaces for each team , and ask them to share modules across workspaces.
B. Implement a Private module registry in Terraform cloud , and ask teams to reference them.
C. Upgrade to Terraform enterprise , since this is not possible in terraform cloud.
D. Upload the modules in the terraform public module registry , and ask teams to reference them

Answer: B

Explanation:
Terraform Cloud's private module registry helps you share Terraform modules across your organization. It includes support for module versioning, a searchable and filterable list of available modules, and a configuration designer to help you build new workspaces faster.
By design, the private module registry works much like the public Terraform Registry. If you're already used the public registry, Terraform Cloud's registry will feel familiar.
Understand the different offerings in Terraform OS, Terraform Cloud and Terraform Enterprise. Terraform Cloud's private module registry helps you share Terraform modules across your organization.
https://www.terraform.io/docs/cloud/registry/index.html https://www.terraform.io/docs/cloud/registry/publish.html

NEW QUESTION 147
- (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 148
- (Exam Topic 3)
Ric wants to enable detail logging and he wants highest verbosity of logs. Which of the following environment variable settings is correct option for him to select.

A. Set TF_LOG = DEBUG
B. Set VAR_TF = TRACE
C. Set TF_LOG = TRACE
D. Set VAR_TF_LOG = TRACE

Answer: C

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

NEW QUESTION 151
- (Exam Topic 3)
In regards to Terraform state file, select all the statements below which are correct?

A. When using local state, the state file is stored in plain-text.
B. The state file is always encrypted at rest.
C. Storing state remotely can provide better security.
D. Using the mask feature, you can instruct Terraform to mask sensitive data in the state file.
E. The Terraform state can contain sensitive data, therefore the state file should be protected from unauthorized access.
F. Terraform Cloud always encrypts state at rest.

Answer: ACEF

Explanation:
Terraform state 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.
When using remote state, state is only ever held in memory when used by Terraform. It may be encrypted at rest, but this depends on the specific remote state backend.
Storing Terraform state remotely can provide better security. As of Terraform 0.9, Terraform does not persist state to the local disk when remote state is in use, and some backends can be configured to encrypt the state data at rest.
Recommendations
If you manage any sensitive data with Terraform (like database passwords, user passwords, or private keys), treat the state itself as sensitive data.
Storing state remotely can provide better security. As of Terraform 0.9, Terraform does not persist state to the local disk when remote state is in use, and some

backends can be configured to encrypt the state data at rest.
For example:
* Terraform Cloud always encrypts state at rest and protects it with TLS in transit. Terraform Cloud also knows the identity of the user requesting state and maintains a history of state changes. This can be used to control access and track activity. Terraform Enterprise also supports detailed audit logging.
* The S3 backend supports encryption at rest when the encrypt option is enabled. IAM policies and logging can be used to identify any invalid access. Requests for the state go over a TLS connection.

NEW QUESTION 153
- (Exam Topic 3)
Which of the following Terraform commands will automatically refresh the state unless supplied with additional flags or arguments? Choose TWO correct answers.

A. terraform state
B. terraform apply
C. terraform plan
D. terraform validate
E. terraform output

Answer: BC

NEW QUESTION 157
- (Exam Topic 3)
Terraform Cloud always encrypts state at rest and protects it with TLS in transit. Terraform Cloud also knows the identity of the user requesting state and maintains a history of state changes.

A. False
B. True

Answer: B

Explanation:
Terraform Cloud always encrypts state at rest and protects it with TLS in transit. Terraform Cloud also knows the identity of the user requesting state and maintains a history of state changes. This can be used to control access and track activity. Terraform Enterprise also supports detailed audit logging.
https://www.terraform.io/docs/state/sensitive-data.html#recommendations

NEW QUESTION 160
- (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 161
- (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 164
- (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 169
- (Exam Topic 3)
Terraform-specific settings and behaviors are declared in which configuration block type?

A. provider
B. terraform
C. resource
D. data

Answer: B

Explanation:
The special terraform configuration block type is used to configure some behaviors of Terraform itself, such as requiring a minimum Terraform version to apply your configuration.
Example terraform {
required_version = "> 0.12.0"
}
https://www.terraform.io/docs/configuration/terraform.html

NEW QUESTION 173
- (Exam Topic 3)
A single terraform resource file that defines an aws_instance resource can simply be renamed to vsphere_virtual_machine in order to switch cloud providers.

A. True
B. False

Answer: B

Explanation:
Every provider has its own required and allowed declarations none of which match between cloud providers.

NEW QUESTION 178
- (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 182
- (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 184
- (Exam Topic 3)
During a terraform apply, a resource is successfully created but eventually fails during provisioning. What happens to the resource?

A. The resource will be planned for destruction and recreation upon the next terraform apply
B. Terraform will retry to provision again.
C. The failure of provisioner will be ignored and it will not cause a failure to terraform apply
D. The resource will be automatically destroyed.

Answer: A

Explanation:
If a creation-time provisioner fails, the resource is marked as tainted. A tainted resource will be planned for destruction and recreation upon the next terraform apply. Terraform does this because a failed provisioner can leave a resource in a semi-configured state. Because Terraform cannot reason about what the provisioner does, the only way to ensure proper creation of a resource is to recreate it. This is tainting.
You can change this behavior by setting the on_failure attribute, which is covered in detail below. https://www.terraform.io/docs/provisioners/index.html#creation-time-provisioners https://www.terraform.io/docs/provisioners/index.html#destroy-time-provisioners https://www.terraform.io/docs/provisioners/index.html#failure-behavior

NEW QUESTION 186
- (Exam Topic 4)
Which of the following is an invalid variable name?

A. count
B. web
C. var1
D. instance_name

Answer: A

Explanation:
https://www.terraform.io/intro/examples/count.html

NEW QUESTION 188
- (Exam Topic 4)
You have decided to create a new Terraform workspace to deploy a development environment. What is different about this workspace?

A. It uses a different branch of code It uses a different backend
B. It has its own state file
C. It pulls in a different terraform.tvars file

Answer: C

NEW QUESTION 189
- (Exam Topic 4)
HashiCorp offers multiple versions of Terraform, including Terraform open-source, Terraform Cloud, and Terraform Enterprise. Which of the following Terraform features are only available in the Enterprise edition? (select four)

A. SAML/SSO
B. Sentinel
C. Audit Logs
D. Clustering
E. Private Module Registry
F. Private Network Connectivity

Answer: ACF

Explanation:

While there are a ton of features that are available to open source users, many features that are part of the Enterprise offering are geared towards larger teams and enterprise functionality. To see what specific features are part of Terraform Cloud and Terraform Enterprise, check out this link.
https://www.hashicorp.com/products/terraform/pricing/

NEW QUESTION 191
- (Exam Topic 4)
Select the most accurate statement to describe the Terraform language from the following list.

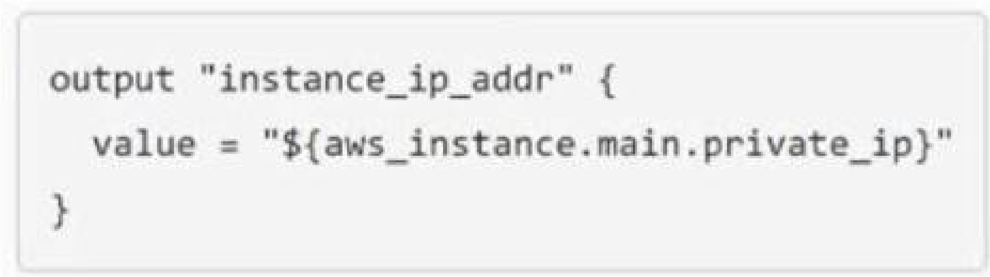
A. Terraform is an immutable, declarative, Infrastructure as Code provisioning language based on Hashicorp Configuration Language, or optionally JSON.
B. Terraform is a mutable, declarative, Infrastructure as Code configuration management language based on Hashicorp Configuration Language, or optionally JSON.
C. Terraform is an immutable, procedural, Infrastructure as Code configuration management language based on Hashicorp Configuration Language, or optionally JSON.
D. Terraform is a mutable, procedural, Infrastructure as Code provisioning language based on Hashicorp Configuration Language, or optionally YAML.

Answer: A

Explanation:
Terraform is not a configuration management tool - https://www.terraform.io/intro/vs/chefpuppet.html Terraform is a declarative language - https://www.terraform.io/docs/configuration/index.html Terraform supports a syntax that is JSON compatible https://www.terraform.io/docs/configuration/syntax-json.html
Terraform is primarily designed on immutable infrastructure principles - https://www.hashicorp.com/resources/what-is-mutable-vs-immutable-infrastructure

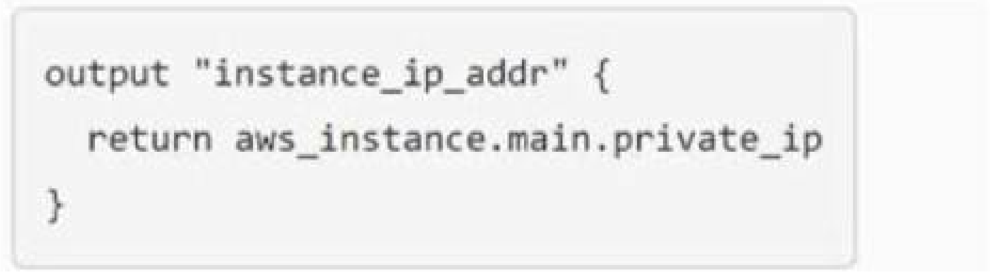
NEW QUESTION 192
- (Exam Topic 4)
You are writing a child Terraform module which provisions an AWS instance. You want to make use of the IP address returned in the root configuration. You name the instance resource "main".
Which of these is the correct way to define the output value using HCL2?

A.



```
output "instance_ip_addr" {
  value = "${aws_instance.main.private_ip}"
}
```

B.



```
output "instance_ip_addr" {
  return aws_instance.main.private_ip
}
```

A. Option A
B. Option B

Answer: A

NEW QUESTION 193
- (Exam Topic 4)
A user runs terraform init on their RHEL based server and per the output, two provider plugins are downloaded: $ terraform init
Initializing the backend... Initializing provider plugins...
- Checking for available provider plugins...
- Downloading plugin for provider "aws" (hashicorp/aws) 2.44.0...
- Downloading plugin for provider "random" (hashicorp/random) 2.2.1...
:
Terraform has been successfully initialized! Where are these plugins downloaded to?

A. The .terraform.plugins directory in the directory terraform init was executed in.
B. The .terraform/plugins directory in the directory terraform init was executed in.
C. /etc/terraform/plugins
D. The .terraform.d directory in the directory terraform init was executed in.

Answer: B

NEW QUESTION 197
- (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 199
- (Exam Topic 4)
Which of the following value will be accepted for my_var?
* 1. variable "my_var"
* 2. {
* 3. type = string
* 4. }

A. 15
B. "15"
C. Both A and B
D. None of the above

Answer: C

Explanation:
The Terraform language will automatically convert number and bool values to string values when needed, and vice-versa as long as the string contains a valid representation of a number or boolean value. Example
* true converts to "true", and vice-versa
* false converts to "false", and vice-versa

* 15 converts to "15", and vice-versa
Where possible, Terraform automatically converts values from one type to another in order to produce the expected type. If this isn't possible, Terraform will produce a type mismatch error and you must update the
configuration with a more suitable expression. https://www.terraform.io/docs/configuration/expressions.html#type-conversion

NEW QUESTION 204
- (Exam Topic 4)
Provider dependencies are created in several different ways. Select the valid provider dependencies from the following list: (select three)

A. Explicit use of a provider block in configuration, optionally including a version constraint.
B. Use of any resource belonging to a particular provider in a resource or data block in configuration.
C. Existence of any resource instance belonging to a particular provider in the current state.
D. Existence of any provider plugins found locally in the working directory.

Answer: ABC

Explanation:
The existence of a provider plugin found locally in the working directory does not itself create a provider dependency. The plugin can exist without any reference to it in the terraform configuration. https://www.terraform.io/docs/commands/providers.html

NEW QUESTION 209
- (Exam Topic 4)
When using constraint expressions to signify a version of a provider, which of the following are valid provider versions that satisfy the expression found in the following code snippet: (select two)
* 1. terraform
* 2. {
* 3. required_providers
* 4. {
* 5. aws = "~> 1.2.0"
* 6. }
* 7. }

A. 1.3.1
B. 1.2.3
C. 1.2.9
D. 1.3.0

Answer: BC

Explanation:
As your Terraform usage becomes more advanced, there are some cases where you may need to modify the Terraform state. Rather than modify the state directly, the terraform state commands can be used in many cases instead. This command is a nested subcommand, meaning that it has further subcommands. https://www.terraform.io/docs/commands/state/index.html

NEW QUESTION 210
- (Exam Topic 4)
Which of the following arguments are required when declaring a Terraform output?

A. sensitive
B. description
C. default
D. value

Answer: D

NEW QUESTION 215
- (Exam Topic 4)
Given the below resource configuration - resource "aws_instance" "web" { # ... count = 4 }
What does the terraform resource address aws_instance.web refer to?

A. It refers to all 4 web instances , together , for further individual segregation , indexing is required , with a 0 based index.
B. It refers to the last web EC2 instance , as by default , if no index is provided , the last / N-1 index is used.
C. It refers to the first web EC2 instance out of the 4 ,as by default , if no index is provided , the first / 0th index is used.
D. The above will result in a syntax error , as it is not syntactically correct . Resources defined using count , can only be referenced using indexes.

Answer: A

Explanation:
A Resource Address is a string that references a specific resource in a larger infrastructure. An address is made up of two parts:
[module path][resource spec] Module path:
A module path addresses a module within the tree of modules. It takes the form: module.A.module.B.module.C...
Multiple modules in a path indicate nesting. If a module path is specified without a resource spec, the address applies to every resource within the module. If the module path is omitted, this addresses the root module.
Given a Terraform config that includes: resource "aws_instance" "web" {
# ...
count = 4
}
An address like this: aws_instance.web[3]
Refers to only the last instance in the config, and an address like this: aws_instance.web

Refers to all four "web" instances. https://www.terraform.io/docs/internals/resource-addressing.html

NEW QUESTION 218
- (Exam Topic 4)
Which parameters does terraform import require? Choose two correct answers.

A. Provider
B. Path
C. Resource address
D. Resource ID

Answer: CD

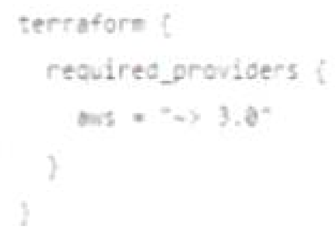
Explanation:
https://www.terraform.io/cli/commands/import#usage

NEW QUESTION 223
- (Exam Topic 4)
Which task does teraform init not perform?

A. Sources any modules and copies the configuration locally
B. Validates all required variables are present
C. Connects to the backend
D. Sources all providers present in the configuration and ensures they are downloaded and available locally

Answer: B

NEW QUESTION 224
- (Exam Topic 4)
What does this code do?



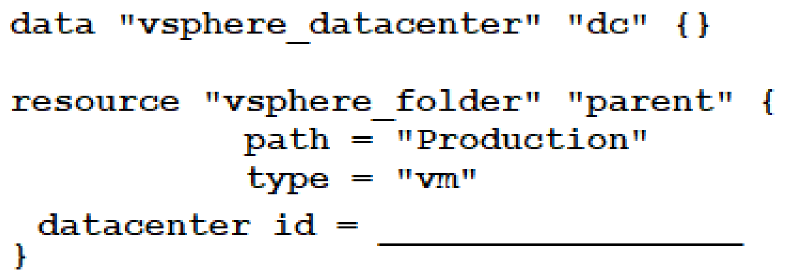
```
terraform {
  required_providers {
    aws = "~> 3.0"
  }
}
```

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 229
- (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 233
- (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 238
- (Exam Topic 4)
Which statements best describes what the local variable assignment is doing in the following code snippet:

A. Create a distinct list of route table name objects
B. Create a map of route table names to subnet names
C. Create a map of route table names from a list of subnet names
D. Create a list of route table names eliminating duplicates

Answer: D

NEW QUESTION 241
- (Exam Topic 4)
A user creates three workspaces from the command line - prod, dev, and test. Which of the following commands will the user run to switch to the dev workspace?

A. terraform workspace dev
B. terraform workspace select dev
C. terraform workspace -switch dev
D. terraform workspace switch dev

Answer: B

Explanation:
The terraform workspace select command is used to choose a different workspace to use for further operations.
https://www.terraform.io/docs/commands/workspace/select.html

NEW QUESTION 245
- (Exam Topic 4)
terraform destroy is the only way to remove infrastructure.

A. True
B. False

Answer: B

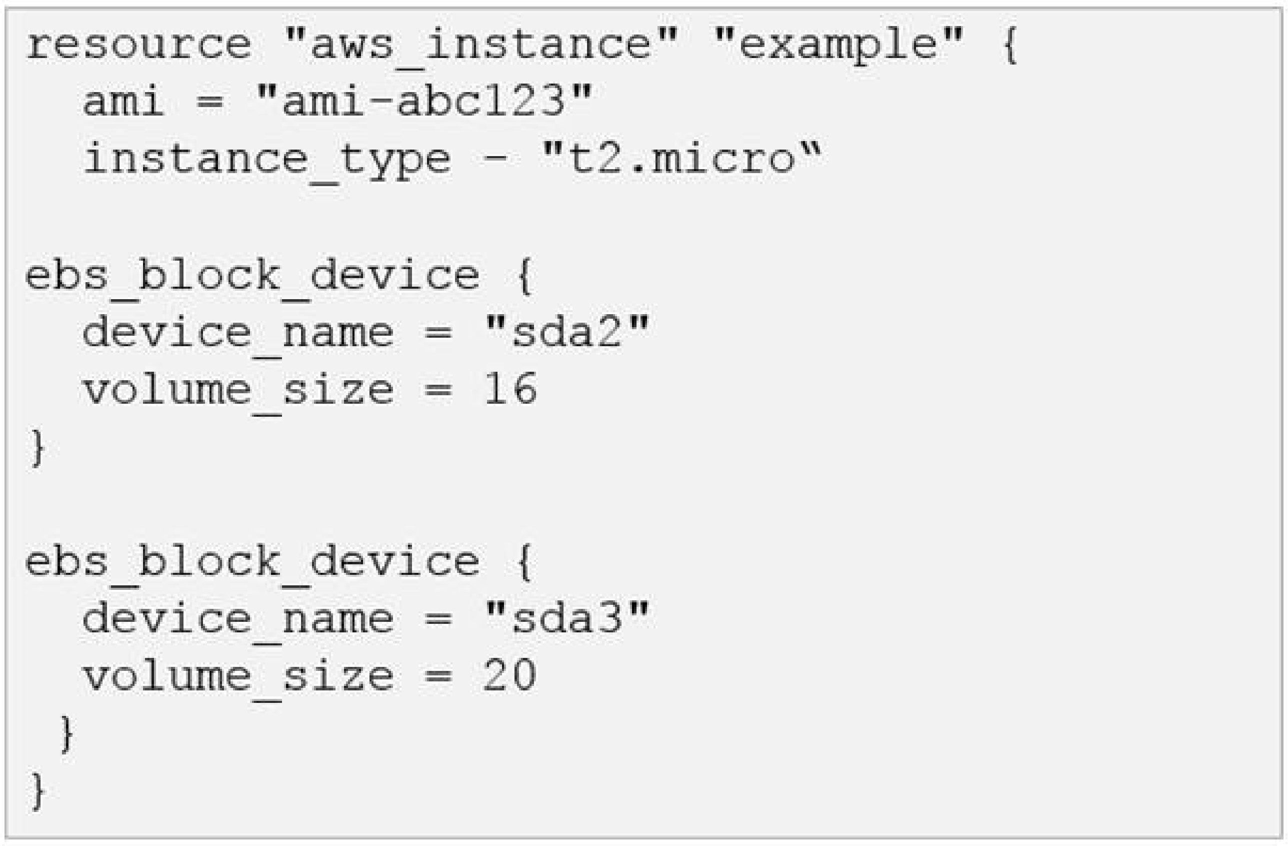
NEW QUESTION 249
- (Exam Topic 4)
You have created a custom variable definition file my_vars.tfvars. How will you use it for provisioning infrastructure?

A. terraform apply -var-state-file ="my_vars.tfvars"
B. terraform apply var-file="my_vars.tfvars"
C. terraform plan -var-file="my_vars.tfvar"
D. terraform apply -var-file="my_vars.tfvars"

Answer: D

Explanation:
To set lots of variables, it is more convenient to specify their values in a variable definitions file (with a filename ending in either .tfvars or .tfvars.json) and then specify that file on the command line with -var-file:
terraform apply -var-file="my_vars.tfvars" https://www.terraform.io/docs/configuration/variables.html#variable-definitions-tfvars-files

NEW QUESTION 253
- (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 258
- (Exam Topic 4)
What feature of Terraform Cloud and/or Terraform Enterprise can you publish and maintain a set of custom modules which can be used within your organization?

A. Terraform registry
B. custom VCS integration
C. private module registry
D. remote runs

Answer: C

NEW QUESTION 263
- (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 266
- (Exam Topic 4)
In a Terraform Cloud workspace linked to a version control repository, speculative plan runs start automatically when you merge or commit changes to version control.

A. True
B. False

Answer: B

NEW QUESTION 267
- (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 269
- (Exam Topic 4)
Terraform will sync all resources in state by default for every plan and apply, hence for larger infrastructures this can slow down terraform plan and terraform apply commands?

A. False
B. True

Answer: B

Explanation:
For small infrastructures, Terraform can query your providers and sync the latest attributes from all your resources. This is the default behavior of Terraform: for every plan and apply, Terraform will sync all resources in your state.
For larger infrastructures, querying every resource is too slow. Many cloud providers do not provide APIs to query multiple resources at once, and the round trip time for each resource is hundreds of milliseconds. On top of this, cloud providers almost always have API rate limiting so Terraform can only request a certain number of resources in a period of time. Larger users of Terraform make heavy use of the -refresh=false flag as well as the -target flag in order to work around this. In these scenarios, the cached state is treated as the record of truth.
https://www.terraform.io/docs/state/purpose.html

NEW QUESTION 270
- (Exam Topic 4)
Which of the following statements about local modules is incorrect:

A. Local modules are not cached by terraform init command
B. Local modules are sourced from a directory on disk
C. Local modules support versions
D. All of the above (all statements above are incorrect
E. None of the above (all statements above are correct)

Answer: C

Explanation:
Version constraints are supported only for modules installed from a module registry, such as the public Terraform Registry or Terraform Cloud's private module registry. Other module sources can provide their own versioning mechanisms within the source string itself, or might not support versions at all. In particular, modules sourced from local file paths do not support version; since they're loaded from the same source repository, they always share the same version as their caller.
https://www.terraform.io/language/modules/syntax

NEW QUESTION 271
- (Exam Topic 4)
Select two answers to complete the following sentence: Before a new provider can be used, it must be _______ and _______.

A. approved by HashiCorp
B. uploaded to source control
C. declared in the configuration
D. initialized

Answer: CD

Explanation:
Each time a new provider is added to configuration -- either explicitly via a provider block or by adding a resource from that provider -- Terraform must initialize the provider before it can be used. Initialization downloads and installs the provider's plugin so that it can later be executed.

NEW QUESTION 274
- (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 276
- (Exam Topic 4)
If a Terraform creation-time provisioner fails, what will occur by default?

A. The resource will not be affected, but the provisioner will need to be applied again
B. The resource will be destroyed
C. The resource will be marked as "tainted"
D. Nothing, provisioners will not show errors in the command line

Answer: C

Explanation:
If a creation-time provisioner fails, the resource is marked as tainted. A tainted resource will be planned for destruction and recreation upon the next terraform apply .

NEW QUESTION 277
- (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 279
- (Exam Topic 4)
Which of the following is not supported backend types in Terra form?

A. consul
B. gcs
C. manta
D. bitbucket

Answer: D

NEW QUESTION 280
- (Exam Topic 4)
Terraform configuration (including any module references) can contain only one Terraform provider type.

A. True
B. False

Answer: B

NEW QUESTION 285
- (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 289
- (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 291
- (Exam Topic 4)
From the answers below, select the advantages of using Infrastructure as Code.

A. Provide a codified workflow to develop customer-facing applications.
B. Safely test modifications using a "dry run" before applying any actual changes.
C. Easily integrate with application workflows (GitLab Actions, Azure DevOps, CI/CD tools).
D. Easily change and update existing infrastructure.
E. Provide reusable modules for easy sharing and collaboration.

Answer: BCDE

Explanation:
Infrastructure as Code is not used to develop applications, but it can be used to help deploy or provision those applications to a public cloud provider or on-premises infrastructure.
All of the others are benefits to using Infrastructure as Code over the traditional way of managing infrastructure, regardless if it's public cloud or on-premises.

NEW QUESTION 292
- (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 293
- (Exam Topic 4)
Which of the following is a meta-argument defined in the configuration files of Terraform?

A. tfvar
B. depends_on
C. instance aws
D. varl

Answer: B

NEW QUESTION 294
- (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 299
- (Exam Topic 4)
You're writing a Terraform configuration that needs to read input from a local file called id_rsa.pub. Which built-in Terraform function can you use to import the file's contents as a string?

A. fileset("id_rsa.pub")
B. filebase64("id_rsa.pub")
C. templatefile("id_rsa.pub")
D. file("id_rsa.pub")

Answer: D

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

NEW QUESTION 300
- (Exam Topic 4)
Changing the Terraform backend from the default "local" backend to a different one after doing your first terraform apply is:

A. Mandatory
B. Optional
C. Impossible
D. Discouraged

Answer: B

NEW QUESTION 305
- (Exam Topic 4)
What kind of configuration block will create an infrastructure object with settings specified in the block?

A. state
B. provider
C. resource
D. data

Answer: C

NEW QUESTION 307
- (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 310
- (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 314
- (Exam Topic 4)
Terraform plan updates your state file.

A. True
B. False

Answer: B

Explanation:
The terraform plan command creates an execution plan, which lets you preview the changes that Terraform plans to make to your infrastructure. The plan command alone will not actually carry out the proposed changes, and so you can use this command to check whether the proposed changes match what you expected before you apply the changes or share your changes with your team for broader review. Source: https://www.terraform.io/cli/commands/plan

NEW QUESTION 319
- (Exam Topic 4)
Suppose terraformcode is taking up some values which are not defined inside the code files. In which of the following options issue might have occurred?

A. Issue in main.tf file
B. Issue in vars.tf file
C. Issue in terraform.tfvars
D. Issue in Environment Variables

Answer: D

NEW QUESTION 321
- (Exam Topic 4)
Which are examples of infrastructure as code? (Choose two.)

A. Cloned virtual machine images
B. Change management database records
C. Versioned configuration files
D. Docker files

Answer: CD

NEW QUESTION 326
- (Exam Topic 4)
Consider the following Terraform 0.12 configuration snippet:
* 1. variable "vpc_cidrs" {
* 2. type = map
* 3. default = {
* 4. us-east-1 = "10.0.0.0/16"
* 5. us-east-2 = "10.1.0.0/16"
* 6. us-west-1 = "10.2.0.0/16"
* 7. us-west-2 = "10.3.0.0/16"
* 8. }
* 9. }
* 10.
* 11. resource "aws_vpc" "shared" {
* 12. cidr_block = ______
* 13. }
How would you define the cidr_block for us-east-1 in the aws_vpc resource using a variable?

A. var.vpc_cidrs.0
B. vpc_cidrs["us-east-1"]
C. var.vpc_cidrs["us-east-1"]
D. var.vpc_cidrs[0]

Answer: C

NEW QUESTION 328
- (Exam Topic 4)
During a terraform plan, a resource is successfully created but eventually fails during provisioning. What happens to the resource?

A. Terraform attempts to provision the resource up to three times before exiting with an error
B. the terraform plan is rolled back and all provisioned resources are removed
C. it is automatically deleted
D. the resource is marked as tainted

Answer: D

Explanation:
If a resource successfully creates but fails during provisioning, Terraform will error and mark the resource as "tainted". A resource that is tainted has been physically created, but can't be considered safe to use since provisioning failed. Terraform also does not automatically roll back and destroy the resource during the apply when the failure happens, because that would go against the execution plan: the execution plan would've said a resource will be created, but does not say it will ever be deleted.

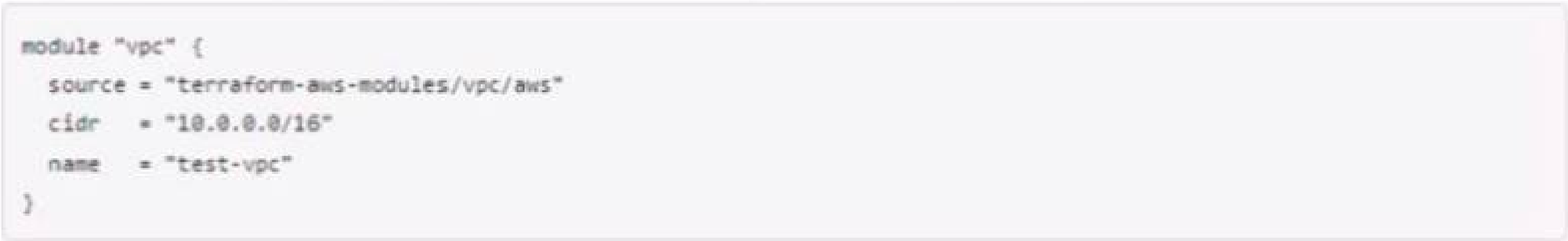
NEW QUESTION 330
- (Exam Topic 4)
When do you need to explicitly execute terraform refresh?

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 335
- (Exam Topic 4)
In the below configuration, how would you reference the module output vpc_id ?



```
module "vpc" {
  source = "terraform-aws-modules/vpc/aws"
  cidr   = "10.0.0.0/16"
  name   = "test-vpc"
}
```

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:
module.vpc.id

NEW QUESTION 337
- (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 339
- (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 340
- (Exam Topic 4)
When configuring a remote backend in Terraform, it might be a good idea to purposely omit some of the required arguments to ensure secrets and other important data aren't inadvertently shared with others. What are the ways the remaining configuration can be added to Terraform so it can initialize and communicate with the backend? (select three)

A. directly querying HashiCorp Vault for the secrets
B. command-line key/value pairs
C. use the -backend-config=PATH to specify a separate config file
D. interactively on the command line

Answer: BCD

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: https://www.terraform.io/docs/backends/init.html#backend-initialization

NEW QUESTION 342
- (Exam Topic 4)
Which provider authentication method prevents credentials from being stored in the state file?

A. Using environment variables
B. Specifying the login credentials in the provider block
C. Setting credentials as Terraform variables
D. None of the above

Answer: A

NEW QUESTION 347
- (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 349
- (Exam Topic 4)
When you use a remote backend that needs authentication. HashrCorp recommends that you:

A. Push your Tefraform configuration to an encrypted git repository
B. Write the authentication credentials in the Terraform configuration files
C. Use partial configuration to load the authentication credentials outside of the Terraform code
D. Keep the Terraform configuration files in a secret store

Answer: C

Explanation:
We recommend omitting the token from the configuration, and instead using terraform login or manually configuring credentials in the CLI config file. Reference: https://www.terraform.io/language/settings/backends/remote

NEW QUESTION 350
- (Exam Topic 4)
terraform validate reports HCL syntax errors.

A. True
B. False

Answer: A

NEW QUESTION 355
- (Exam Topic 4)
What advantage does an operations team that uses infrastructure as code have?

A. The ability to delete infrastructure
B. The ability to reuse best practice configurations and settings
C. The ability to autoscale a group of servers
D. The ability to update existing infrastructure

Answer: B

NEW QUESTION 356
- (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 358
- (Exam Topic 4)
Which of the following is not an advantage of using infrastructure as code operations?

A. Self-service infrastructure deployment
B. Troubleshoot via a Linux diff command
C. Public cloud console configuration workflows
D. Modify a count parameter to scale resources
E. API driven workflows

Answer: B

Explanation:
terraform is used to deploy the infrastructure, not to troubleshoot it

NEW QUESTION 359
- (Exam Topic 4)
You have a Terraform configuration that defines a single virtual machine with no references to it. You have run terraform apply to create the resource, and then removed the resource definition from your Terraform configuration file.
What will happen when you run terraform apply in the working directory again?

A. Nothing
B. Terraform will destroy the virtual machine
C. Terraform will error
D. Terraform will remove the virtual machine from the state file, but the resource will still exist

Answer: B

Explanation:
If you remove the resource from your config file and the resource is in your state file, terraform will apply the configuration in the config file - which is to delete the resource

NEW QUESTION 363
- (Exam Topic 4)
Select the operating systems which are supported for a clustered Terraform Enterprise: (select four)

A. Unix
B. Red Hat
C. CentOS
D. Amazon Linux
E. Ubuntu

Answer: BCDE

Explanation:

https://www.terraform.io/docs/enterprise/before-installing/index.html#operating-systemrequirements

NEW QUESTION 367
- (Exam Topic 4)
Terraform Cloud is available only as a paid offering from HashiCorp.

A. True
B. False

Answer: B

Explanation:
Many of Terraform Cloud features are free for small teams, including remote state storage, remote runs, and VCS connections.
"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."

NEW QUESTION 370
- (Exam Topic 4)
What does Terraform use .terraform.lock.hcl file for?

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

Answer: A

Explanation:
https://www.terraform.io/language/files/dependency-lock
"hcl , and this name is intended to signify that it is a lock file for various items that Terraform caches in the . terraform subdirectory of your working directory. Terraform automatically creates or updates the dependency lock file each time you run the terraform init command."

NEW QUESTION 374
- (Exam Topic 4)
Your team lead does not trust the junior terraform engineers who now have access to the git repo . So , he wants you to have some sort of a checking layer , whereby , you can ensure that the juniors will not create any non-compliant resources that might lead to a security audit failure in future. What can you do to efficiently enforce this?

A. Create a design /security document (in PDF) and share to the team , and ask them to always follow that document , and never deviate from it.
B. Since your team is using Hashicorp Terraform Enterprise Edition , enable Sentinel , and writePolicy-As-Code rules that will check for non-compliant resource provisioning , and prevent/report them.
C. Use Terraform OSS Sentinel Lite version , which will save cost , since there is no charge for OSS , but it can still check for most non-compliant rules using Policy-As-Code.
D. Create a git master branch , and implement PR . Every change needs to be reviewed by you , before being merged to the master branch.

Answer: B

Explanation:
Sentinel is an embedded policy-as-code framework integrated with the HashiCorp Enterprise products. It enables fine-grained, logic-based policy decisions, and can be extended to use information from external sources.
https://www.terraform.io/docs/cloud/sentinel/index.html

NEW QUESTION 378
- (Exam Topic 4)
Which of the following is not valid source path for specifying a module?

A. source = "./modulelversion=v1.0.0"
B. source = "github.com/hashicorp/example?ref=v1.0.0"
C. source = "./module"
D. source = "hashicorp/consul/aws"

Answer: A

NEW QUESTION 379
- (Exam Topic 4)
You need to migrate a workspace to use a remote backend. After updating your configuration, what command do you run to perform the migration?
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:
Once you have authenticated to Terraform Cloud, you're ready to migrate your local state file to Terraform Cloud. To begin the migration, reinitialize. This causes Terraform to recognize your cloud block configuration.

NEW QUESTION 384
- (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 385
- (Exam Topic 4)
Select all features which are exclusive to Terraform Enterprise. (Select Three)

A. Sentinel
B. Cost Estimation
C. Audit Logs
D. Clustering
E. SAML/SSO

Answer: CDE

Explanation:
Sentinel and Cost Estimation are also available in Terraform Cloud https://www.hashicorp.com/products/terraform/pricing/

NEW QUESTION 386
- (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 388
- (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 393
- (Exam Topic 4)
What Terraform feature is shown in the example below?

A. conditional expression
B. local values
C. dynamic block
D. data source

Answer: C

NEW QUESTION 398
- (Exam Topic 4)
Which of the following is not a valid Terraform string function?

A. replace
B. format
C. join
D. tostring

Answer: D

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

NEW QUESTION 402
- (Exam Topic 4)
Which of the below backends support state locking?

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

Answer: ABC

NEW QUESTION 405
- (Exam Topic 4)
When writing Terraform code, HashiCorp recommends that you use how many spaces between each nesting level?

A. 1
B. 2
C. 4

Answer: C

Explanation:
The Terraform parser allows you some flexibility in how you lay out the elements in your configuration files, but the Terraform language also has some idiomatic style conventions which we recommend users always follow for consistency between files and modules written by different teams. Automatic source code formatting tools may apply these conventions automatically.
Indent two spaces for each nesting level.
When multiple arguments with single-line values appear on consecutive lines at the same nesting level, align their equals signs:
ami = "abc123" instance_type = "t2.micro"
When both arguments and blocks appear together inside a block body, place all of the arguments together at the top and then place nested blocks below them. Use one blank line to separate the arguments from the blocks.
Use empty lines to separate logical groups of arguments within a block.
For blocks that contain both arguments and "meta-arguments" (as defined by the Terraform language semantics), list meta-arguments first and separate them from other arguments with one blank line. Place meta-argument blocks last and separate them from other blocks with one blank line.

```
resource "aws_instance" "example" { count = 2 # meta-argument first
ami = "abc123" instance_type = "t2.micro" network_interface {
# ...
}
lifecycle { # meta-argument block last create_before_destroy = true
}
}
```

Top-level blocks should always be separated from one another by one blank line. Nested blocks should also be separated by blank lines, except when grouping together related blocks of the same type (like multiple provisioner blocks in a resource).
Avoid separating multiple blocks of the same type with other blocks of a different type, unless the block types are defined by semantics to form a family. (For example: root_block_device, ebs_block_device and
ephemeral_block_device on aws_instance form a family of block types describing AWS block devices, and can therefore be grouped together and mixed.)

NEW QUESTION 406
- (Exam Topic 4)
As a member of an operations team that uses infrastructure as code (laC) practices, you are tasked with making a change to an infrastructure stack running in a public cloud. Which pattern would follow laC 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 411
- (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 415
......