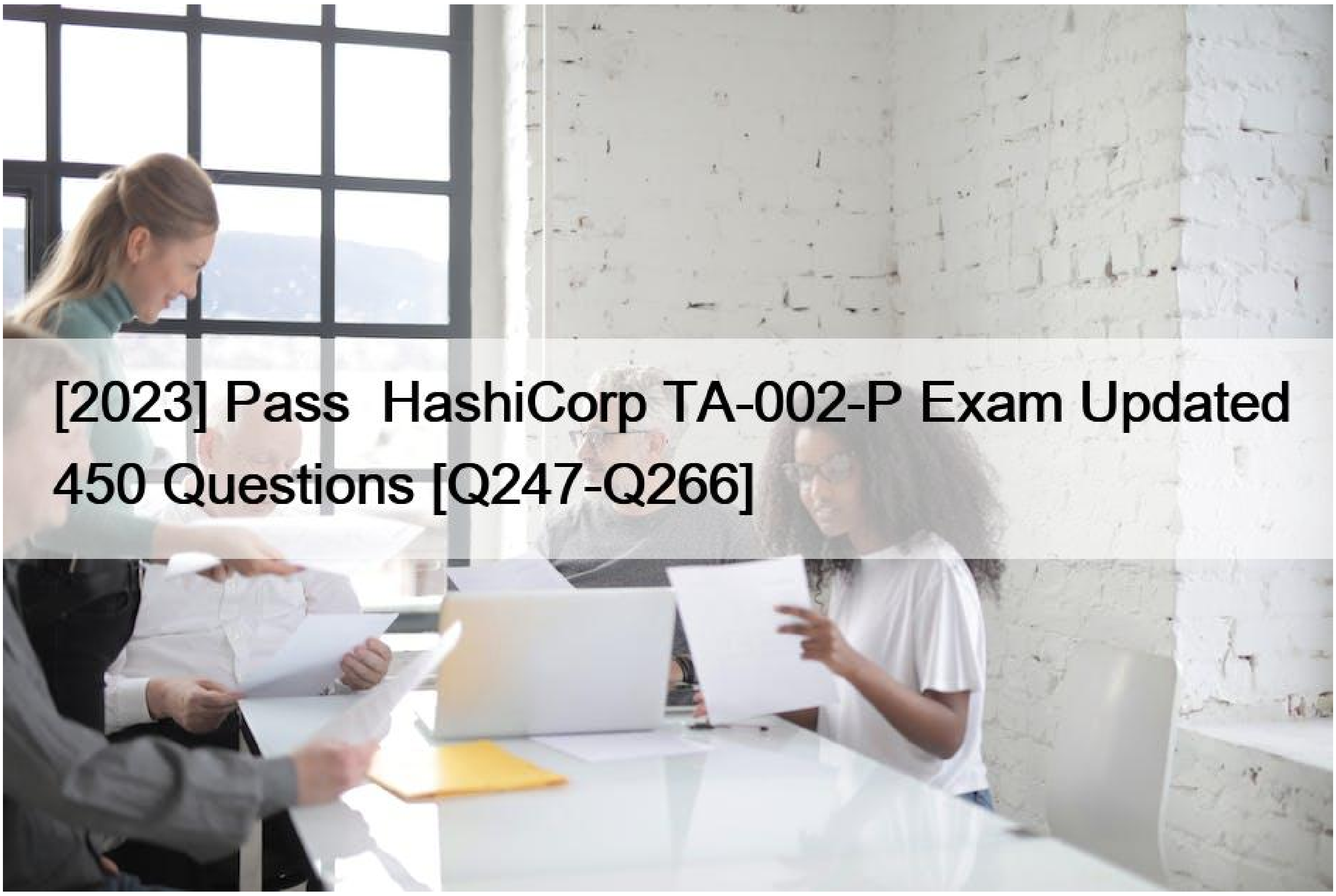
[2023 Pass HashiCorp TA-002-P Exam Updated 450 Questions [Q247-Q266



[2023] Pass HashiCorp TA-002-P Exam Updated 450 Questions
Get 2023 Updated Free HashiCorp TA-002-P Exam Questions and Answer

NEW QUESTION 247

terraform validate validate validates that your infrastructure matches the Terraform state file.
* True
* False
Explanation

The terraform validate command validates the configuration files in a directory, referring only to the configuration and not accessing any remote services such as remote state, provider APIs, etc. Validate runs checks that verify whether a configuration is syntactically valid and internally consistent, regardless of any provided variables or existing state. It is thus primarily useful for general verification of reusable modules, including correctness of attribute names and value types. Source:

https://www.terraform.io/cli/commands/validate

NEW QUESTION 248

Setting the TF_LOG environment variable to DEBUG causes debug messages to be logged into syslog.
* True
* False
Explanation/Reference: https://www.terraform.io/docs/internals/debugging.html

NEW QUESTION 249

If you enable TF_LOG = DEBUG, the log will be stored in syslog.log file in the currect directory.
* False
* True
https://www.terraform.io/docs/internals/debugging.html

NEW QUESTION 250

You have recently started a new job at a retailer as an engineer. As part of this new role, you have been tasked with evaluating multiple outages that occurred during peak shopping time during the holiday season.

Your investigation found that the team is manually deploying new compute instances and configuring each compute instance manually. This has led to inconsistent configuration between each compute instance.

How would you solve this using infrastructure as code?
* Implement a ticketing workflow that makes engineers submit a ticket before manually provisioning and configuring a resource
* Implement a checklist that engineers can follow when configuring compute instances
* Replace the compute instance type with a larger version to reduce the number of required deployments
* Implement a provisioning pipeline that deploys infrastructure configurations committed to your version control system following code reviews

NEW QUESTION 251

What does the command terraform fmt do?
* Rewrite Terraform configuration files to a canonical format and style.
* Deletes the existing configuration file.
* Updates the font of the configuration file to the official font supported by HashiCorp.
* Formats the state file in order to ensure the latest state of resources can be obtained.
The terraform fmt command is used to rewrite Terraform configuration files to a canonical format and style. This command applies a subset of the Terraform language style conventions, along with other minor adjustments for readability.

Other Terraform commands that generate Terraform configuration will produce configuration files that conform to the style imposed by terraform fmt, so using this style in your own files will ensure consistency.

https://www.terraform.io/docs/commands/fmt.html

NEW QUESTION 252

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.)
* Run terraform plan -destroy.
* This is not possible. You can only show resources that will be created.
* Run terraform state rm *.
* Run terraform destroy and it will first output all the resources that will be deleted before prompting for

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

NEW QUESTION 253

The terraform init command is always safe to run multiple times, to bring the working directory up to date

with changes in the configuration. Though subsequent runs may give errors, this command will never delete

your existing configuration or state.
* False
* True
Explanation

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

NEW QUESTION 254

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

NEW QUESTION 255

In contrast to Terraform Open Source, when working with Terraform Enterprise and Cloud Workspaces, conceptually you could think about them as completely separate working directories.
* True
* False

NEW QUESTION 256

Module version is required to reference a module on the Terraform Module Registry.
* True
* False

NEW QUESTION 257

Your configuration file has been locked accidentally. What of the following command would you use to unlock?
* terraform filename-unlock
* delete the file and create a new state file
* terraform force-unlock
* state.tf-unlock

NEW QUESTION 258

You have been given requirements to create a security group for a new application. Since your organization

standardizes on Terraform, you want to add this new security group with the fewest number of lines of code.

What feature could you use to iterate over a list of required tcp ports to add to the new security group?
* dynamic backend
* splat expression
* terraform import
* dynamic block

Explanation

A dynamic block acts much like a for expression, but produces nested blocks instead of a complex typed

value. It iterates over a given complex value and generates a nested block for each element of that complex

value.

https://www.terraform.io/docs/configuration/expressions.html#dynamic-blocks

NEW QUESTION 259

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)
* SAML/SSO
* Sentinel
* Audit Logs
* Clustering
* Private Module Registry
* Private Network Connectivity

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 260

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?
* terraform unlock
* terraform force-unlock
* terraform state unlock
* None of the above
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 261

You have provisioned some virtual machines (VMs) on Google Cloud Platform (GCP) using the gcloud

command line tool. However, you are standardizing with Terraform and want to manage these VMs using

Terraform instead.

What are the two things you must do to achieve this? (Choose two.)
* Provision new VMs using Terraform with the same VM names
* Use the terraform import command for the existing VMs
* Write Terraform configuration for the existing VMs
* Run the terraform import-gcp command
Explanation

You should create the equivalent configuration first, and then run import to load it on the state file.

NEW QUESTION 262

After executing a terraform apply, you notice that a resource has a tilde (~) next to it. What does this infer?
* The resource will be updated in place.
* The resource will be created.
* Terraform can&#8217;t determine how to proceed due to a problem with the state file.
* The resource will be destroyed and recreated.

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&#8230; [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 &#8220;aws_instance&#8221; &#8220;example&#8221; {

~ ami = &#8220;ami-2757f631&#8221; -> &#8220;ami-b374d5a5&#8221; # forces replacement

~ arn = &#8220;arn:aws:ec2:us-east-1:130490850807:instance/i-0bbf06244e44211d1&#8221; -> (known after apply)

~ associate_public_ip_address = true -> (known after apply)

~ availability_zone = &#8220;us-east-1c&#8221; -> (known after apply)

~ cpu_core_count = 1 -> (known after apply)

~ cpu_threads_per_core = 1 -> (known after apply)

&#8211; disable_api_termination = false -> null

&#8211; ebs_optimized = false -> null

get_password_data = false

+ host_id = (known after apply)

~ id = &#8220;i-0bbf06244e44211d1&#8221; -> (known after apply)

~ instance_state = &#8220;running&#8221; -> (known after apply)

instance_type = &#8220;t2.micro&#8221;

~ ipv6_address_count = 0 -> (known after apply)

~ ipv6_addresses = [] -> (known after apply)
```

```
+ key_name = (known after apply)
&#8211; monitoring = false -> null
+ network_interface_id = (known after apply)
+ password_data = (known after apply)
+ placement_group = (known after apply)
~ primary_network_interface_id = &#8220;eni-0f1ce5bdae258b015&#8221; -> (known after apply)
~ private_dns = &#8220;ip-172-31-61-141.ec2.internal&#8221; -> (known after apply)
~ private_ip = &#8220;172.31.61.141&#8221; -> (known after apply)
~ public_dns = &#8220;ec2-54-166-19-244.compute-1.amazonaws.com&#8221; -> (known after apply)
~ public_ip = &#8220;54.166.19.244&#8221; -> (known after apply)
~ security_groups = [
&#8211; &#8220;default&#8221;,
] -> (known after apply)
source_dest_check = true
~ subnet_id = &#8220;subnet-1facdf35&#8221; -> (known after apply)
~ tenancy = &#8220;default&#8221; -> (known after apply)
~ volume_tags = {} -> (known after apply)
~ vpc_security_group_ids = [
&#8211; &#8220;sg-5255f429&#8221;,
] -> (known after apply)
&#8211; credit_specification {
&#8211; cpu_credits = &#8220;standard&#8221; -> 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 = &#8220;vol-0079e485d9e28a8e5&#8221; -> (known after apply)
~ volume_size = 8 -> (known after apply)
~ volume_type = &#8220;gp2&#8221; -> (known after apply)
}
```

}

Plan: 1 to add, 0 to change, 1 to destroy.

NEW QUESTION 263

Which of the following statements about Terraform modules is not true?
* Modules must be publicly accessible
* Modules can be called multiple times
* Module is a container for one or more resources
* Modules can call other modules
Explanation

In addition to modules from the local filesystem, Terraform can load modules from a public or private registry.

Also, members of your organization might produce modules specifically crafted for your own infrastructure

needs. Source: https://www.terraform.io/language/modules

NEW QUESTION 264

What allows you to conveniently switch between multiple instances of a single configuration within its single

backend?
* Local backends
* Providers
* Remote backends
* Workspaces
Explanation

Named workspaces allow conveniently switching between multiple instances of a single configuration within

its single backend. &#8230; 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

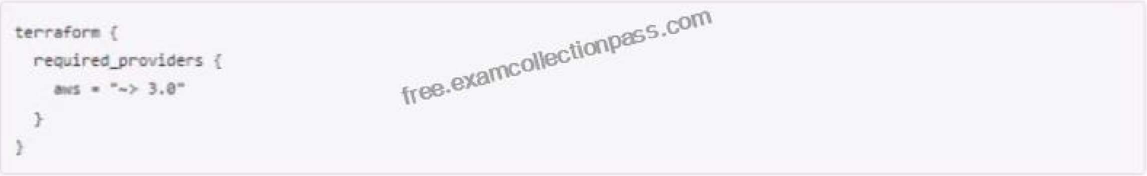
only one backend, but multiple distinct instances of that configuration to be deployed without configuring a

new backend or changing authentication credentials.

https://www.terraform.io/docs/state/workspaces.html

NEW QUESTION 265

What does this code do?



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

* Requires any version of the AWS provider > = 3.0 and < 4.0
* Requires any version of the AWS provider > = 3.0
* Requires any version of the AWS provider after the 3.0 major release like 4.1
* Requires any version of the AWS provider > 3.0

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 266

A Terraform output that sets the &#8221;sensitive&#8221; argument to true will not store that value in the state file.

* True
* False

Verified TA-002-P exam dumps Q&As with Correct 450 Questions and Answers:
https://www.examcollectionpass.com/HashiCorp/TA-002-P-practice-exam-dumps.html]