

TA-002-P^{Q&As}

HashiCorp Certified: Terraform Associate

Pass HashiCorp TA-002-P Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.leads4pass.com/ta-002-p.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by HashiCorp
Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers



QUESTION 1

After creating a new workspace "PROD" you need to run the command terraform select PROD to switch to it.

A. False

B. True

Correct Answer: A

By default, when you create a new workspace you are automatically switched to it To create a new workspace and switch to it, you can use terraform workspace new ; to switch to a existing workspace you can use

terraform workspace select ;

Example:

```
$ terraform workspace new example
```

Created and switched to workspace "example"!

You\\re now on a new, empty workspace. Workspaces isolate their state, so if you run "terraform plan" Terraform will not see any existing state for this configuration.

QUESTION 2

Multiple provider instances blocks for AWS can be part of a single configuration file?

A. False

B. True

Correct Answer: B

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

targeting multiple Docker hosts, multiple Consul hosts, etc.

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

```
# The default provider configuration
```

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

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

```
alias = "west"

region = "us-west-2"

}
```

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

is considered to be the default provider configuration.

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

QUESTION 3

What is the provider for this fictitious resource?

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

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

Correct Answer: C

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

QUESTION 4

All standard backend types support state storage, locking, and remote operations like plan, apply and destroy.

- A. True
- B. False

Correct Answer: B

<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."

QUESTION 5

Valarie has created a database instance in AWS and for ease of use is outputting the value of the database password with the following code. Valarie wants to hide the output value in the CLI after terraform apply that's why she has used sensitive parameter.

1.

```
output "db_password" {
```

2.

```
value = local.db_password
```

3.

```
sensitive = true
```

4.

```
}
```

Since sensitive is set to true, will the value associated with db password be available in plain-text in the state file for everyone to read?

A. Yes

B. No

Correct Answer: A

Outputs can be marked as containing sensitive material by setting the sensitive attribute to true, like this: `output "sensitive" { sensitive = true value = VALUE }` When outputs are displayed on-screen following a terraform apply or terraform refresh, sensitive outputs are redacted, with `displayed` in place of their value. Limitations of Sensitive Outputs The values of sensitive outputs are still stored in the Terraform state, and available using the terraform output command, so cannot be relied on as a sole means of protecting values. Sensitivity is not tracked internally, so if the output is interpolated in another module into a resource, the value will be displayed.

[TA-002-P PDF Dumps](#)[TA-002-P VCE Dumps](#)[TA-002-P Practice Test](#)