

**Good job!**

The code is invalid because it defines two providers of the same type, "aws", with different regions. Terraform requires that each provider be unique. <https://www.terraform.io/language/providers/configuration#alias-multiple-provider-configurations>

Question 1:

Will this code result in an error?

```
1  provider "aws" {  
2    region    = "us-west-2"  
3  }  
4  
5  provider "aws" {  
6    region    = "us-west-1"  
7  }
```

☒ Yes

☐ No



**Good job!**

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

Question 2:

After creating an EC2 instance, Yash wants to automatically install certain software packages like Nginx inside that EC2 instance.

What is the way to achieve it?



Make use of Local provisioner



Make use of Remote provisioner



Make use of Null provisioner



Perform the installation part manually after EC2 instance is created.

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



### Good job!

The "-upgrade" flag tells Terraform to upgrade all providers to the latest version.  
<https://www.terraform.io/cli/commands/init>

Question 3:

There is an outdated provider version in the `.terraform` directory that needs to be updated to the latest version.

Which terraform command can be used to achieve this easily?



terraform plan



terraform upgrade



terraform init -upgrade



Specify the new version under provider block do terraform init

<https://www.terraform.io/cli/commands/init>



### Good job!

The "terraform fmt" command does not write to the Terraform state file, so it can be run safely even if the state file is locked

Question 4:

Matthew is performing some write operation and the Terraform state file is currently locked.

If Bob runs the `terraform fmt` command on the same project, will the action be allowed or blocked?



Allowed



Blocked

Question 5:

Supratik has joined a Security Architect in Enterprise Corp.

He has mandated that all the Terraform configuration that creates an S3 bucket must have encryption feature enabled. What is the best way to achieve it?

- ☐ Review all the S3 terraform code before terraform apply.
- ☐ Create a script that checks if the encryption parameter is enabled on every git commit.
- ☒ **Make use of Sentinel Policies.**
- ☐ Mandate S3 encryption within terraform policies.



**Good job!**

Question 6:

There are 3 resources that have been created using Terraform.

1 resource among them has been deleted manually from the AWS console.

What will happen in the next `terraform apply` operation?

- ☒ **The Deleted resource will be created again.**
- ☐ Terraform will throw an error.
- ☐ No Modification will happen.
- ☐ None of the Above.

**Good job!**

Dynamic block can be useful for managing complex firewall configurations in a more efficient and maintainable way

This was discussed in Lecture 67: [Dynamic Blocks](#) >

Question 7:

Enterprise Corp has recently tied up with multiple small organizations to exchange database information.

Due to this, the firewall rules are increasing, and there are more than 100 rules.

This is a leading firewall configuration file that is difficult to manage.

What is the way this type of configuration can be managed easily?



Terraform Functions



Conditional Expressions



Spat Expressions



Dynamic Blocks



**Good job!**

<https://www.terraform.io/language/settings/backends/remote>

This was discussed in Lecture 134: **Overview of Remote Backends** >

Question 8:

Alice works for Enterprise Corp and they have recently started to use Terraform Cloud.

What is the way in which the log output of terraform plan and apply operation that are running in Terraform Cloud be streamed in the local terminal?

☐ Terraform Modules

☒ Remote Backends

☐ Terraform Cloud Outputs

☐ Terraform Backends



### Good job!

Terraform automatically converts number and bool values to strings when needed.

Question 9:

James has created a variable and has explicitly defined the type as a string. Following is the snippet:

```
1 | variable "myvar" {  
2 |   type = string  
3 | }
```

Which of the following value will be accepted?

☐ 2

☐ "2"

☒ Both of the above values.

☐ None of the above.



**Good job!**

<https://www.terraform.io/cli/commands/taint>

Question 10:

Alice has created 10 different AWS resources from a single terraform configuration file named `large.tf`

Due to some manual modification to one of the EC2 instances, she wants to destroy and recreate the EC2 instance.

What is the ideal way to achieve it?



**terraform destroy**



**terraform recreate**



**terraform taint**



**Manually delete that EC2 instance and run terraform apply that will recreate the EC2 instance.**



**Good job!**

Question 11:

Matthew is currently running a write operation on a terraform project, and the state file is currently locked.

If Bob runs a "terraform apply" command on the same project, will the action be allowed or blocked?



**Allowed**



**Blocked**



**Good job!**

Validate will be allowed since it does not perform any write operation.

Question 12:

Matthew is currently running a write operation on a terraform project, and the state file is currently locked.

If Bob runs a "terraform validate" command on the same project, will the action be allowed or blocked?



**Allowed**



Blocked

**Good job!**

Sentinel Policies are checked when a run is performed, after the terraform plan but before it can be confirmed or the terraform apply is executed.

Question 13:

Matthew has recently subscribed to the Terraform Cloud Trial version.

As part of POC, he is testing the Sentinel feature. At what stage does the Sentinel feature run in Terraform Cloud?



**Before Apply Operation**



After Apply Operation



Before Init Operation



Before Plan Operation.



**Good job!**

<https://www.terraform.io/cli/commands/graph>

Question 14:

Alice wants to create a DOT file in Terraform.

Which terraform command can be used to create it?



**terraform dot**



**terraform dot init**



**terraform graph**



**terraform graphviz**

<https://www.terraform.io/cli/commands/graph>

**Good job!**

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.

Question 15:

There is a requirement to save a terraform plan to a specific file.

Which of the following command can be used to achieve this use-case?

☐ `terraform plan out <filename>`

☐ `terraform output plan <filename>`

☒ `terraform plan -out=<filename>`

☐ `terraform plan <filename>`



Good job!

Question 16:

Enterprise Corp is using the count parameter to create 50 set of IAM users from a specific list.

Following is a sample configuration code:

```
1 | resource "aws_iam_user" "lb" {  
2 |   name = var.elb_names[count.index]  
3 |   count = 50  
4 |   path = "/system/"  
5 | }
```

There is an attribute associated with the resource of aws\_iam\_user named arn.

There is a requirement that the ARN associated with all 50 IAM Users should be shown in the CLI .

What of the following syntax can be used as initials to achieve this?



resource.\*.arn



resource.aws\_iam\_user.lb[\*].arn



resource.arn



\*arn\*



aws\_iam\_user.lb[\*].arn



Good job!

Question 18:

Alice has generated a Terraform plan and saved it to a file that is committed to GIT repository.

Based on this use-case, which of the following statement is true?



Only Alice will be able to perform apply operation based on the generated plan.



Any user who has access can perform apply operation from the generated plan.



**Good job!**

<https://www.terraform.io/language/resources/provisioners/local-exec>

Question 20:

Alice wants to run a specific program in local laptop after Terraform creates a resource.

Which provisioner can be used to achieve this use-case?

☒ **local-exec**

☐ remote-exec

☐ null-exec

☐ file



**Good job!**

Question 21:

John has run the terraform validate command on his project.

What are the type of errors that terraform validate command can report?

☐ Difference between Desired and Current State

☐ Formatting issues in Terraform code.

☐ Deprecated Terraform Blocks.

☒ **Multiple Resource blocks with same local name that are defined twice.**



### Good job!

Overuse of dynamic blocks can make configuration hard to read and maintain, so we recommend using them only when you need to hide details in order to build a clean user interface for a re-usable module.

Question 22:

Why does HashiCorp NOT recommend using dynamic block extensively within Terraform configuration?

What is one of the downsides of using dynamic blocks in a Terraform configuration?



**Extensive use of Dynamic Blocks make core harder for others to read and understand**



Slows down terraform plan operations.



Cannot be used to loop through a list of values



Introduces security issues.



### Good job!

State locking in Terraform is only required for write operations. Read operations, such as terraform state list, can be performed even if the state file is locked.

Question 23:

Matthew is currently running a write operation on a terraform project, and the state file is currently locked.

If Bob runs a `terraform state list` command on the same project, will this action be allowed or blocked?



**Allowed**



Blocked



**Good job!**

<https://www.terraform.io/language/modules/sources>

Question 24:

The Terraform module installer supports installation from a number of different source types, which of these is not a supported source type?



GitHub repository



Terraform Registry



Local path



FTP server



Generic Git, Mercurial repositories

<https://www.terraform.io/language/modules/sources>



**Good job!**

Question 25:

There are 20 resource in AWS created using Terraform configuration. The state file is stored locally.

As part of Terraform best-practice, the team have decided to migrate the `terraform.tfstate` file from local backend to S3 backend.

Which of the following statements are true?



Since the resources are already created, we cannot perform the migration of backend. It is possible only if no resources are created.



Terraform backend can be migrated anytime even if the resources are already created.



Good job!

Question 26:

Developers in Medium Corp are facing a few issues while writing complex expressions that involve interpolation. They have to run the terraform plan every time and check whether there are errors, and also verify the terraform apply to print value as a temporary output for the purpose of debugging

What can be achieved to avoid this?



Use the terraform validate command



Use the terraform console command to run your interpolations and verify and debug them



Use terraform interpolate command.



Use Terraform Cloud as it has dedicated feature to verify the expressions.





Good job!

Question 27:

Matthew has written the following Terraform configuration file. In which order will the resources be created?

```
1 | resource "aws_instance" "myec2" {  
2 |     ami = "ami-12345"  
3 |     instance_type = "t2.large"  
4 | }  
5 |  
6 | resource "aws_eip" "myeip"{  
7 |     vpc = true  
8 |     instance = aws_instance.myec2.id  
9 | }
```



**aws\_eip will be created first**  
**aws\_instance will be created second**



**aws\_eip will be created first**  
**aws\_instance will be created second**



**Resources will be created simultaneously**



**aws\_instance will be created first**  
**aws\_eip will be created second**

**Good job!**

The ~ symbol in the Terraform plan operation indicates that a resource will be updated.  
<https://github.com/hashicorp/terraform/issues/14379>

Question 28:

What does the ~ symbol in terraform plan operation indicate?

```
~ resource "aws_iam_user" "lb" {  
    id          = "demo-user"  
    name        = "demo-user"  
    ~ tags      = {  
        + "tag-key" = "tag-value"  
    }  
    ~ tags_all   = {  
        + "tag-key" = "tag-value"  
    }  
    # (4 unchanged attributes hidden)  
}
```



Addition of New Resource.



Update In Place



Delete in Place



None of the Above.

<https://github.com/hashicorp/terraform/issues/14379>



**Good job!**

<https://www.terraform.io/language/settings>

Question 29:

Matthew needs to configure some behaviors of Terraform, such as requiring a minimum Terraform version to apply your configuration.

Which configuration block can be used to achieve this use-case?

☐ module

☐ provider

☐ resource

☒ terraform

<https://www.terraform.io/language/settings>



**Good job!**

<https://www.terraform.io/cli/commands/workspace/new>

Question 30:

There are a total of 3 workspaces available in a Terraform project.

`workspacea`, `workspaceb`, `workspacec`.

Matthew wants to create an additional workspace named `testing`. Which command will create a new workspace?

☒ **terraform workspace new testing**

☐ terraform workspace -new testing

☐ terraform workspace -create testing

☐ terraform workspace create testing

<https://www.terraform.io/cli/commands/workspace/new>



**Good job!**

<https://www.terraform.io/language/resources/provisioners/connection#example-usage>

Question 31:

Which of the following block allows setting the credentials required while defining Terraform provisioners?

☐ terraform block

☐ credentials block

☒ **connection block**

☐ None of the Above