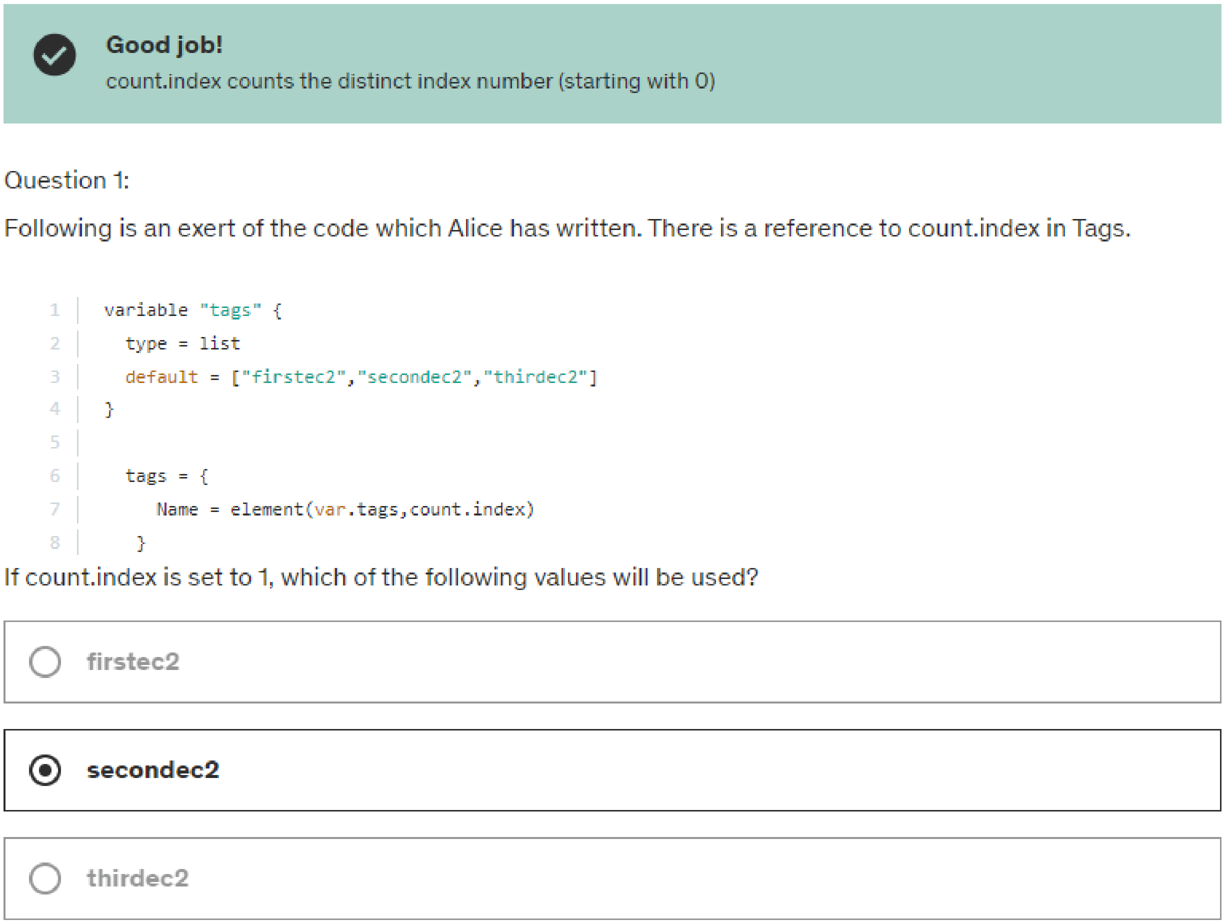


Good job!
count.index counts the distinct index number (starting with 0)

Question 1:

Following is an exert of the code which Alice has written. There is a reference to count.index in Tags.

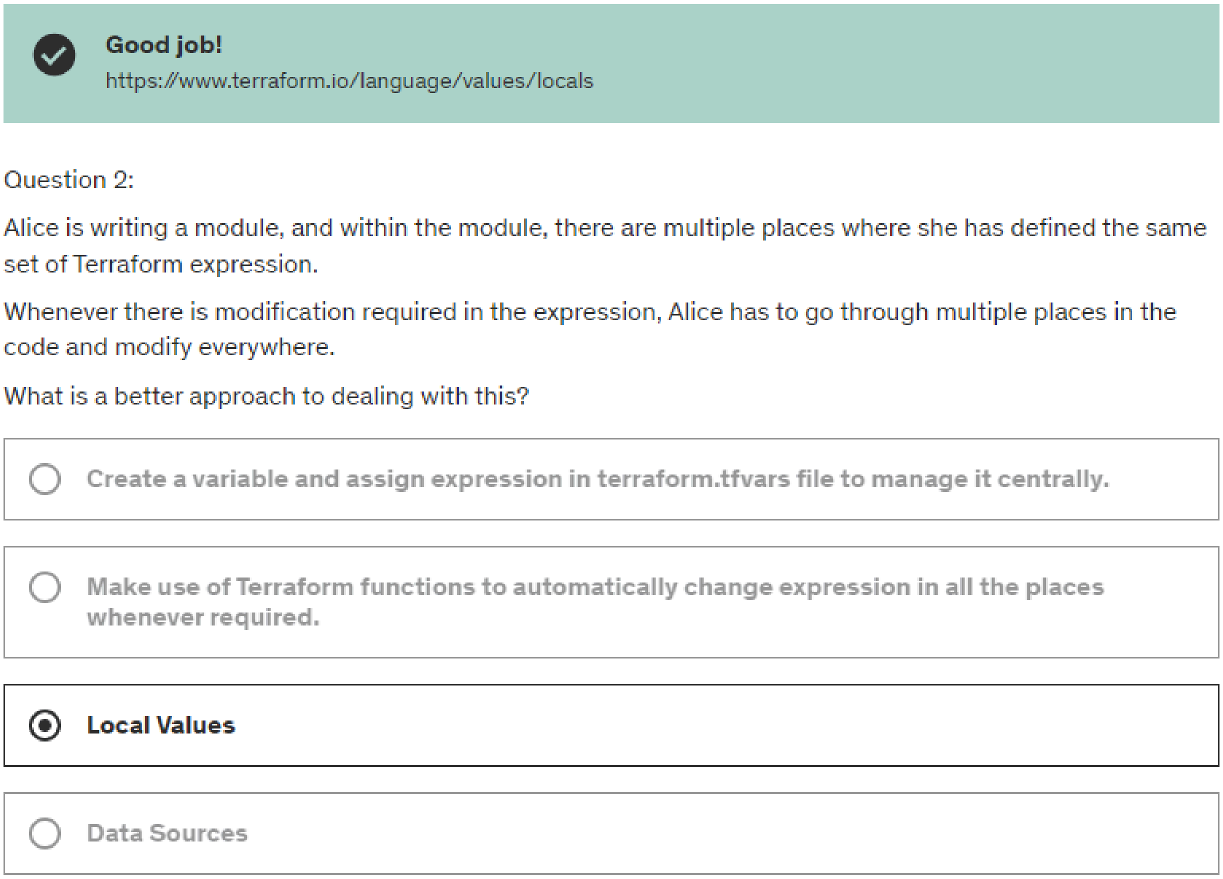
```
variable "tags" {
  type = list
  default = ["firstec2","secondec2","thirdec2"]
}

  tags = {
     Name = element(var.tags,count.index)
   }
```

If count.index is set to 1, which of the following values will be used?

○ firstec2

● **secondec2**

○ thirdec2



✓ **Good job!**

https://www.terraform.io/language/values/locals

Question 2:

Alice is writing a module, and within the module, there are multiple places where she has defined the same set of Terraform expression.

Whenever there is modification required in the expression, Alice has to go through multiple places in the code and modify everywhere.

What is a better approach to dealing with this?

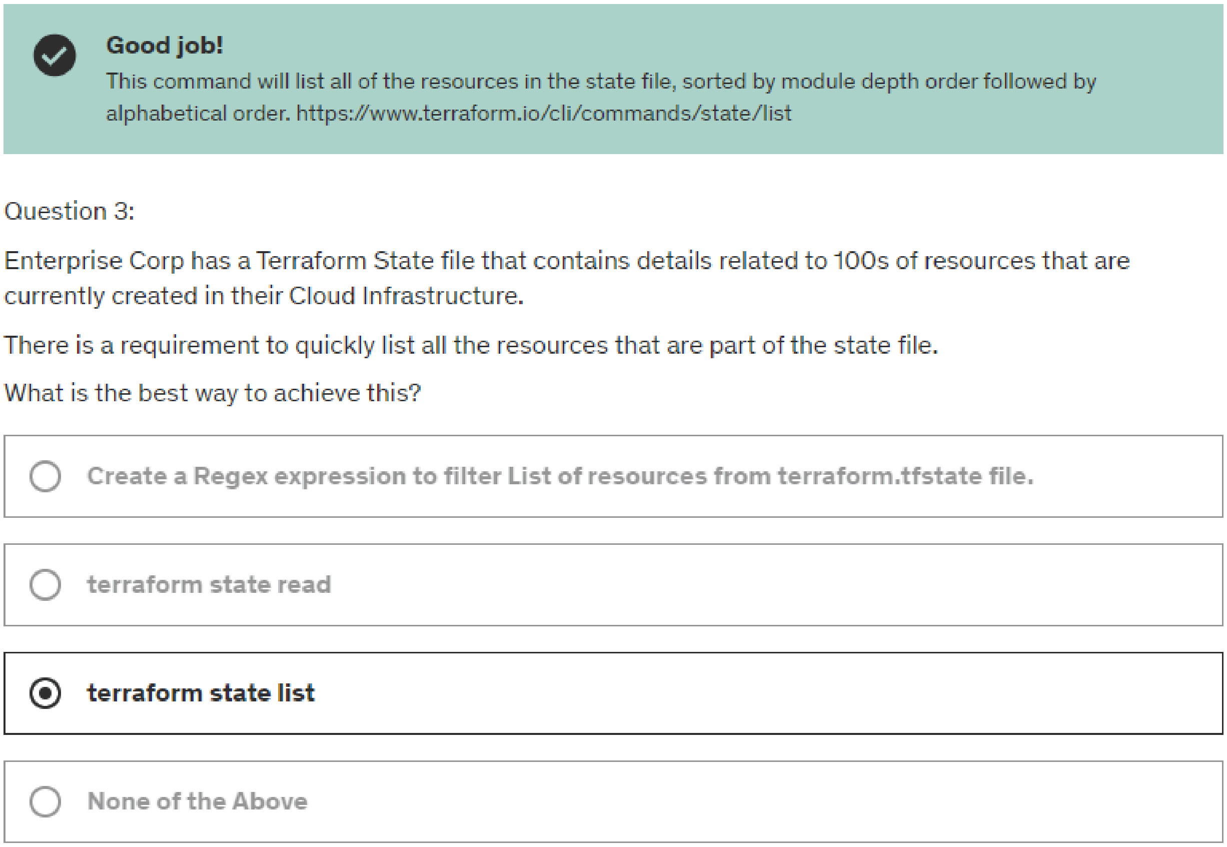
○ **Create a variable and assign expression in terraform.tfvars file to manage it centrally.**

○ **Make use of Terraform functions to automatically change expression in all the places whenever required.**

● **Local Values**

○ **Data Sources**

https://www.terraform.io/language/values/locals



✓ **Good job!**

This command will list all of the resources in the state file, sorted by module depth order followed by alphabetical order. https://www.terraform.io/cli/commands/state/list

Question 3:

Enterprise Corp has a Terraform State file that contains details related to 100s of resources that are currently created in their Cloud Infrastructure.

There is a requirement to quickly list all the resources that are part of the state file.

What is the best way to achieve this?

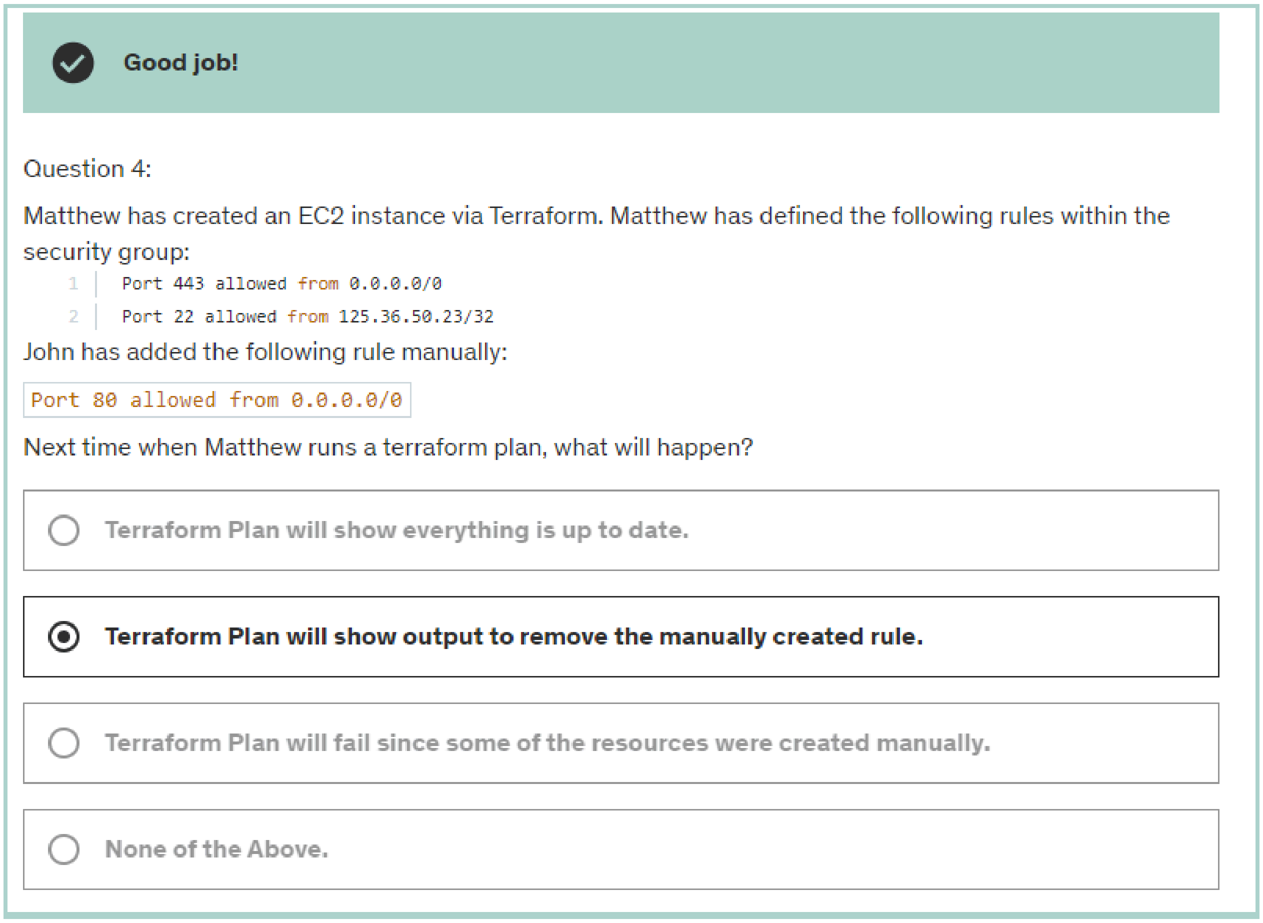
○ Create a Regex expression to filter List of resources from terraform.tfstate file.

○ terraform state read

● **terraform state list**

○ None of the Above

https://www.terraform.io/cli/commands/state/list



✓ **Good job!**

Question 4:

Matthew has created an EC2 instance via Terraform. Matthew has defined the following rules within the security group:

```
Port 443 allowed from 0.0.0.0/0
Port 22 allowed from 125.36.50.23/32
```

John has added the following rule manually:

`Port 80 allowed from 0.0.0.0/0`

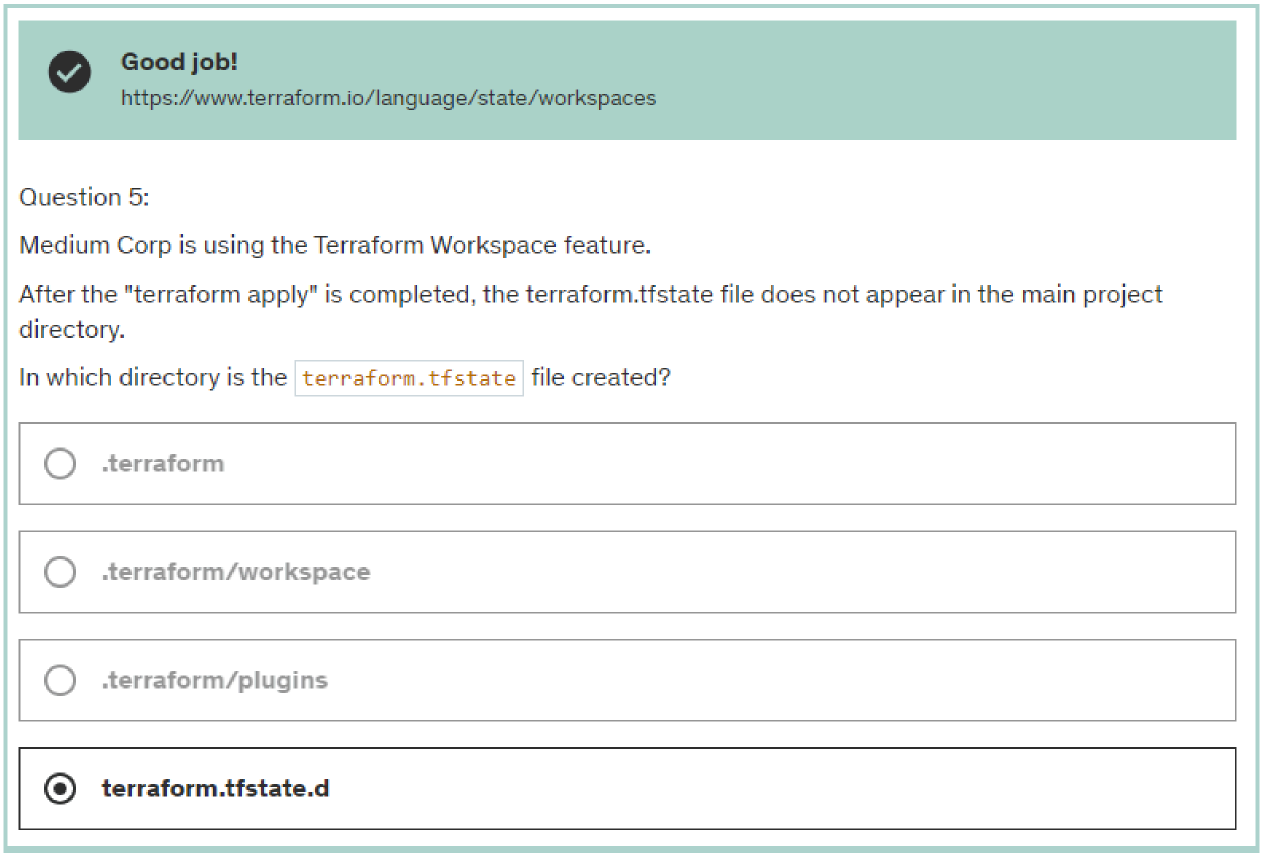
Next time when Matthew runs a terraform plan, what will happen?

○ Terraform Plan will show everything is up to date.

● **Terraform Plan will show output to remove the manually created rule.**

○ Terraform Plan will fail since some of the resources were created manually.

○ None of the Above.



✓ **Good job!**

https://www.terraform.io/language/state/workspaces

Question 5:

Medium Corp is using the Terraform Workspace feature.

After the "terraform apply" is completed, the terraform.tfstate file does not appear in the main project directory.

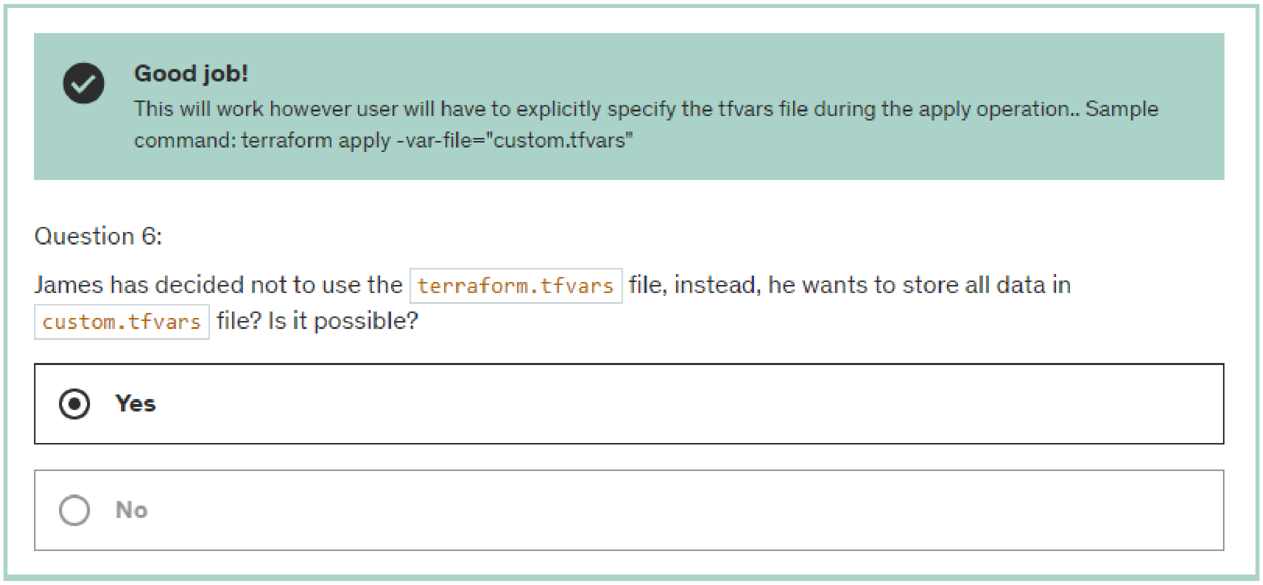
In which directory is the `terraform.tfstate` file created?

○ .terraform

○ .terraform/workspace

○ .terraform/plugins

● **terraform.tfstate.d**



✓ **Good job!**

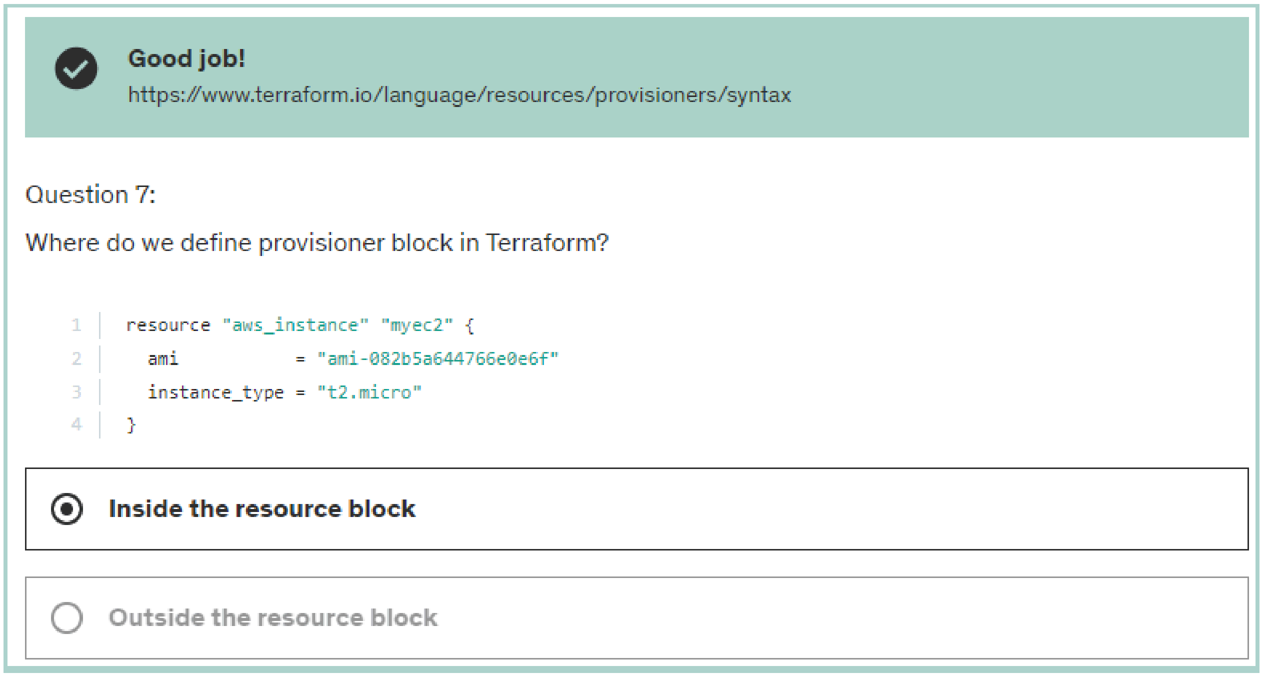
This will work however user will have to explicitly specify the tfvars file during the apply operation.. Sample command: terraform apply -var-file="custom.tfvars"

Question 6:

James has decided not to use the `terraform.tfvars` file, instead, he wants to store all data in `custom.tfvars` file? Is it possible?

● **Yes**

○ No



Good job!

https://www.terraform.io/language/resources/provisioners/syntax

Question 7:

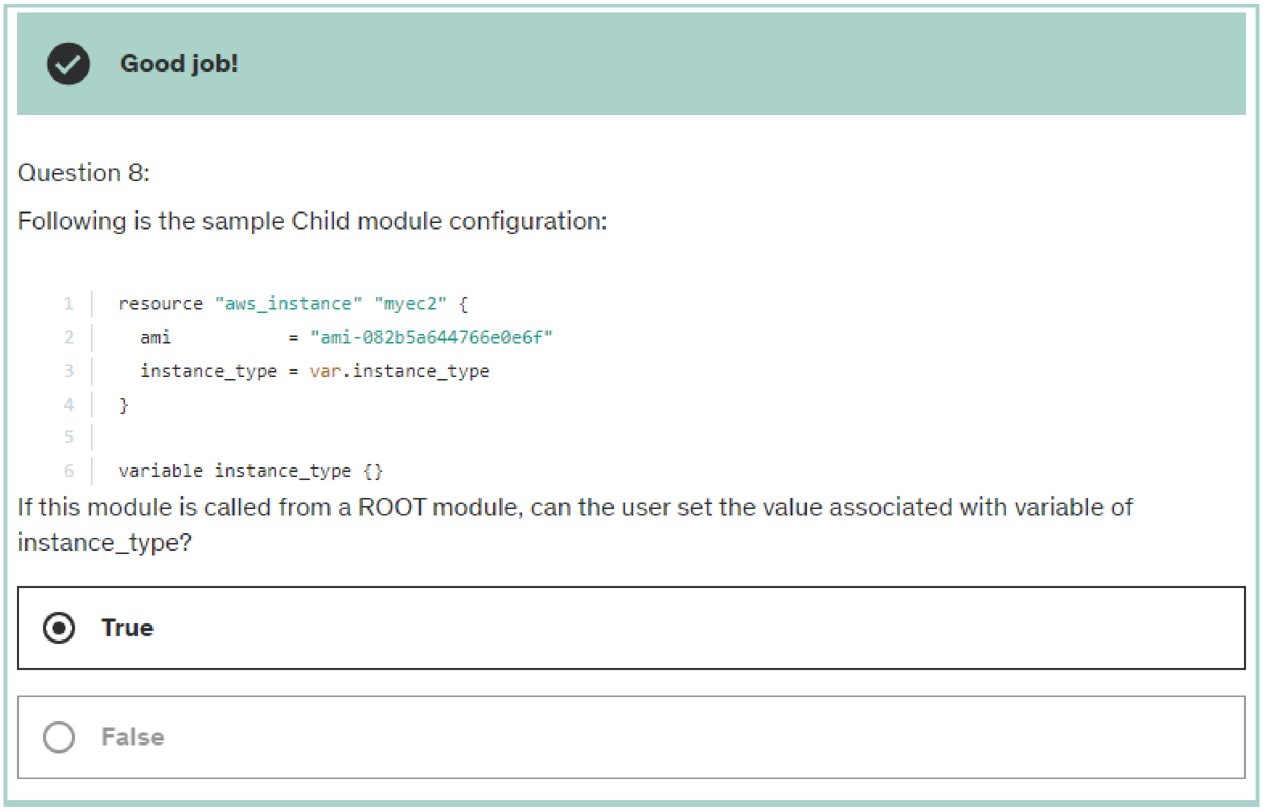
Where do we define provisioner block in Terraform?

```
resource "aws_instance" "myec2" {
  ami           = "ami-082b5a644766e0e6f"
  instance_type = "t2.micro"
}
```

● **Inside the resource block**

○ **Outside the resource block**

https://www.terraform.io/language/resources/provisioners/syntax



Good job!

Question 8:

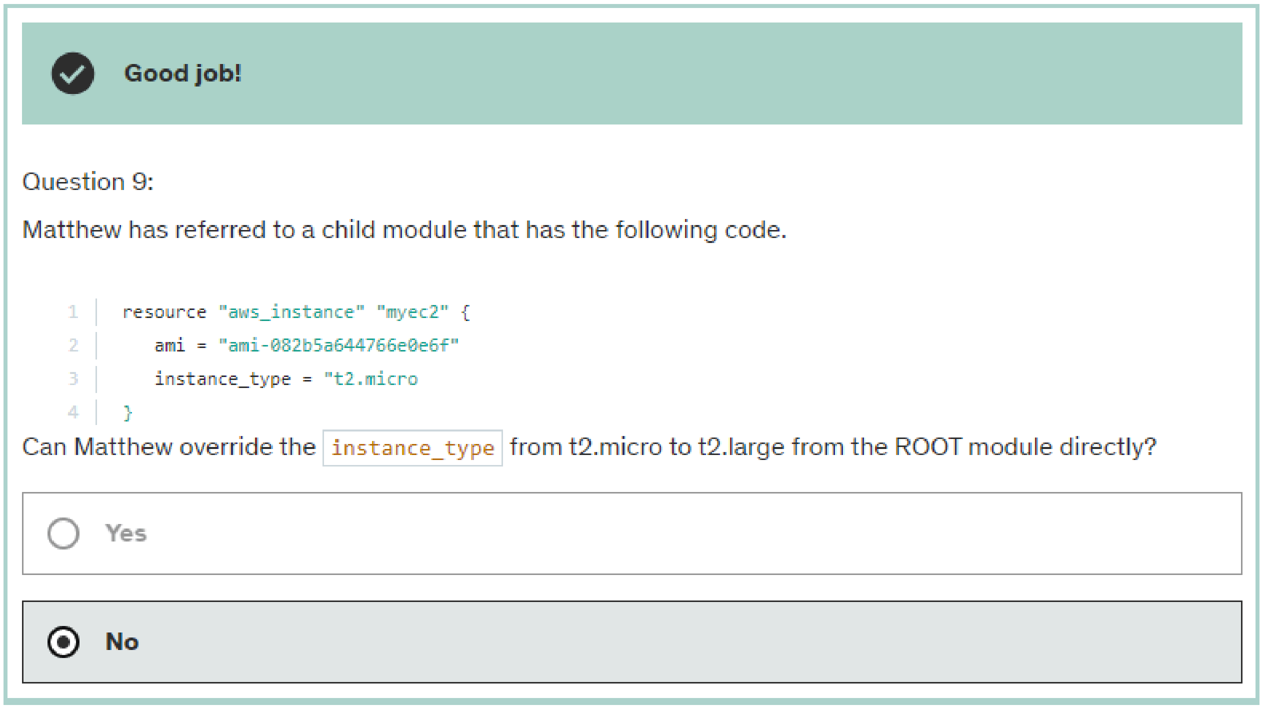
Following is the sample Child module configuration:

```
resource "aws_instance" "myec2" {
  ami           = "ami-082b5a644766e0e6f"
  instance_type = var.instance_type
}

variable instance_type {}
```

If this module is called from a ROOT module, can the user set the value associated with variable of instance_type?

● **True**

○ **False**



✓ **Good job!**

Question 9:

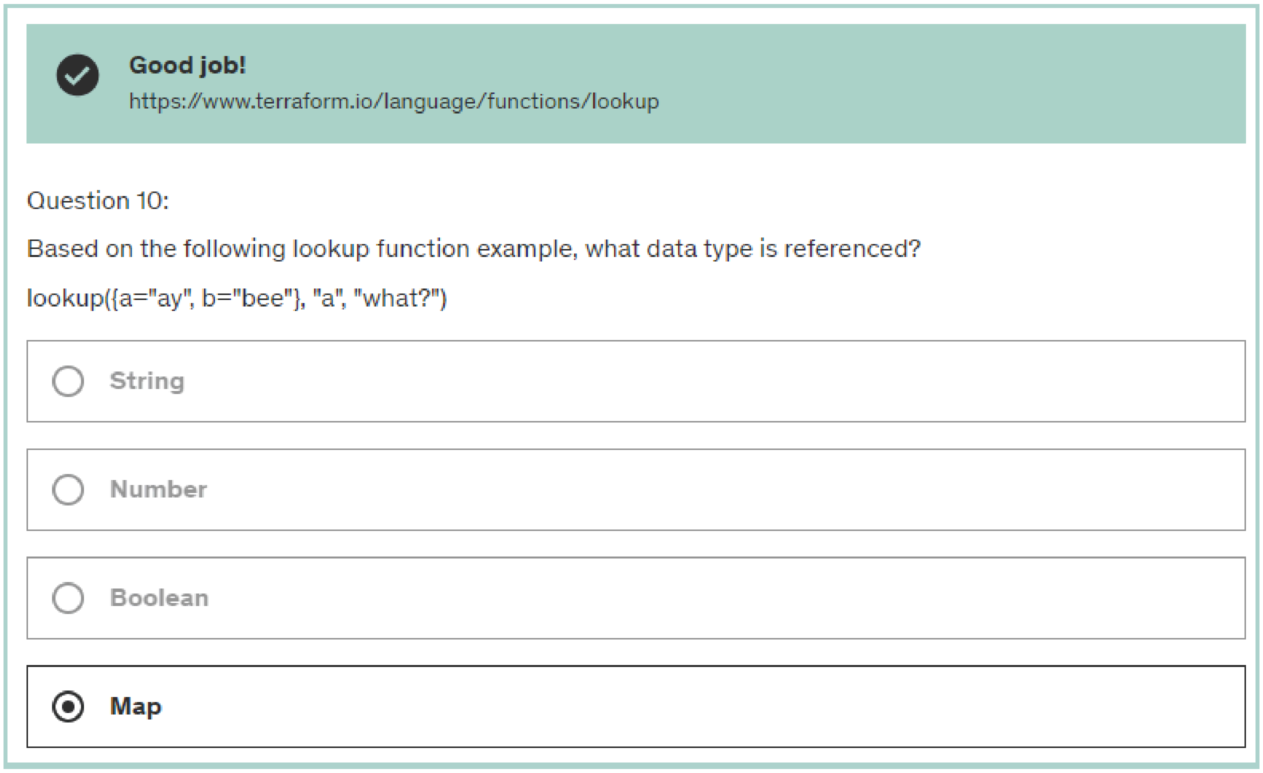
Matthew has referred to a child module that has the following code.

```
resource "aws_instance" "myec2" {
   ami = "ami-082b5a644766e0e6f"
   instance_type = "t2.micro
}
```

Can Matthew override the `instance_type` from t2.micro to t2.large from the ROOT module directly?

○ Yes

● **No**



✓ **Good job!**
https://www.terraform.io/language/functions/lookup

Question 10:

Based on the following lookup function example, what data type is referenced?

lookup({a="ay", b="bee"}, "a", "what?")

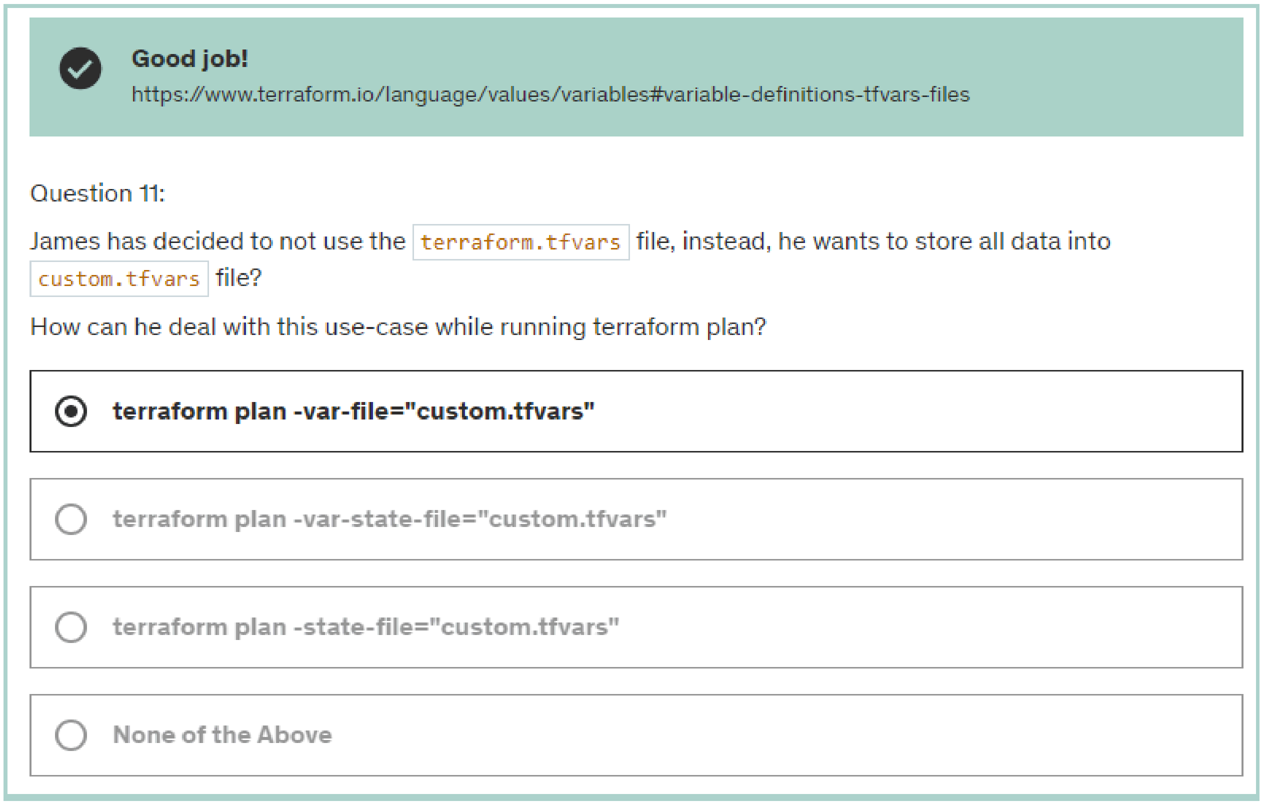
○ String

○ Number

○ Boolean

● **Map**

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



Good job!
https://www.terraform.io/language/values/variables#variable-definitions-tfvars-files

Question 11:

James has decided to not use the `terraform.tfvars` file, instead, he wants to store all data into `custom.tfvars` file?

How can he deal with this use-case while running terraform plan?

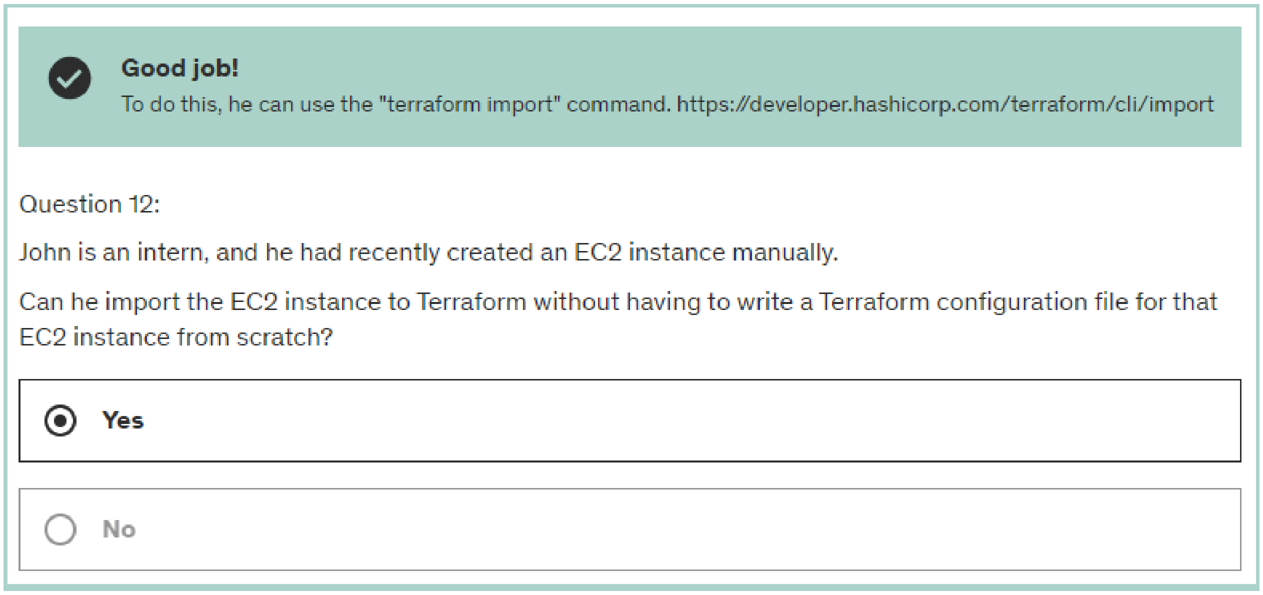
● **terraform plan -var-file="custom.tfvars"**

○ terraform plan -var-state-file="custom.tfvars"

○ terraform plan -state-file="custom.tfvars"

○ None of the Above

https://www.terraform.io/language/values/variables#variable-definitions-tfvars-files



Good job!
To do this, he can use the "terraform import" command. https://developer.hashicorp.com/terraform/cli/import

Question 12:

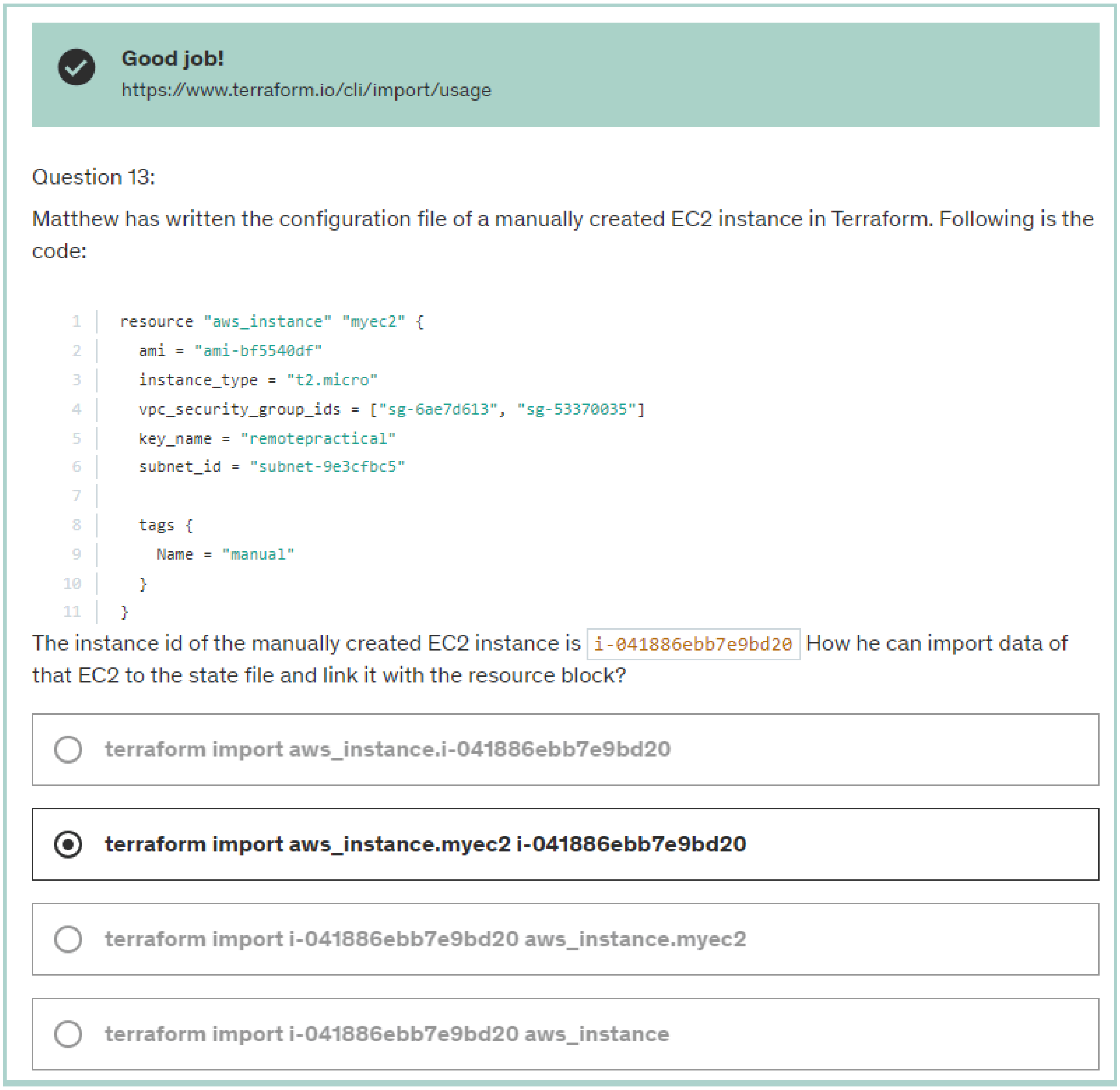
John is an intern, and he had recently created an EC2 instance manually.

Can he import the EC2 instance to Terraform without having to write a Terraform configuration file for that EC2 instance from scratch?

● **Yes**

○ No

https://developer.hashicorp.com/terraform/cli/import



Good job!

https://www.terraform.io/cli/import/usage

Question 13:

Matthew has written the configuration file of a manually created EC2 instance in Terraform. Following is the code:

```
resource "aws_instance" "myec2" {
  ami = "ami-bf5540df"
  instance_type = "t2.micro"
  vpc_security_group_ids = ["sg-6ae7d613", "sg-53370035"]
  key_name = "remotepractical"
  subnet_id = "subnet-9e3cfbc5"

  tags {
    Name = "manual"
  }
}
```

The instance id of the manually created EC2 instance is `i-041886ebb7e9bd20` How he can import data of that EC2 to the state file and link it with the resource block?

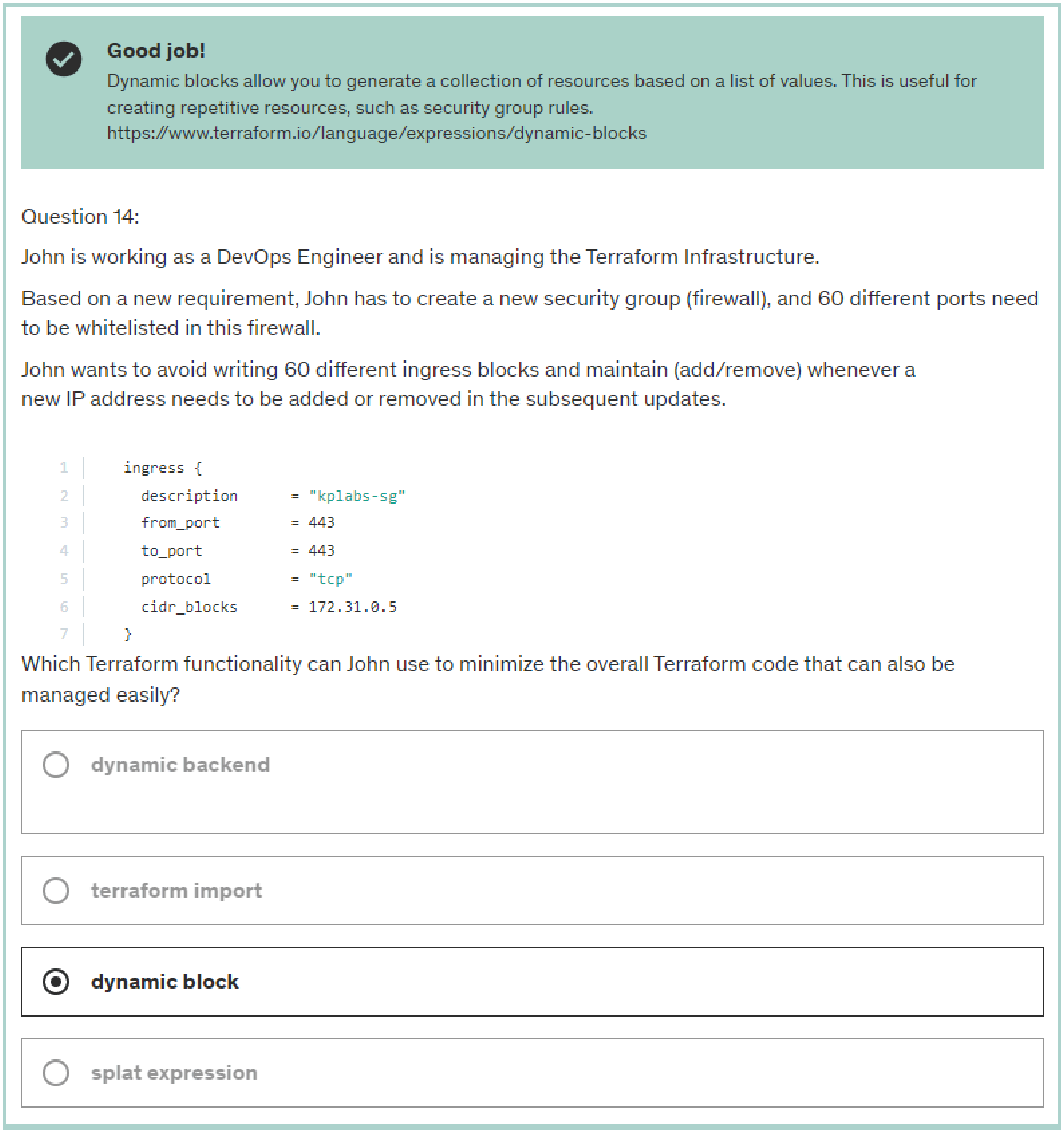
○ terraform import aws_instance.i-041886ebb7e9bd20

● **terraform import aws_instance.myec2 i-041886ebb7e9bd20**

○ terraform import i-041886ebb7e9bd20 aws_instance.myec2

○ terraform import i-041886ebb7e9bd20 aws_instance

https://www.terraform.io/cli/import/usage



Good job!

Dynamic blocks allow you to generate a collection of resources based on a list of values. This is useful for creating repetitive resources, such as security group rules.

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

Question 14:

John is working as a DevOps Engineer and is managing the Terraform Infrastructure.

Based on a new requirement, John has to create a new security group (firewall), and 60 different ports need to be whitelisted in this firewall.

John wants to avoid writing 60 different ingress blocks and maintain (add/remove) whenever a new IP address needs to be added or removed in the subsequent updates.

```
ingress {
  description      = "kplabs-sg"
  from_port        = 443
  to_port          = 443
  protocol         = "tcp"
  cidr_blocks      = 172.31.0.5
}
```

Which Terraform functionality can John use to minimize the overall Terraform code that can also be managed easily?

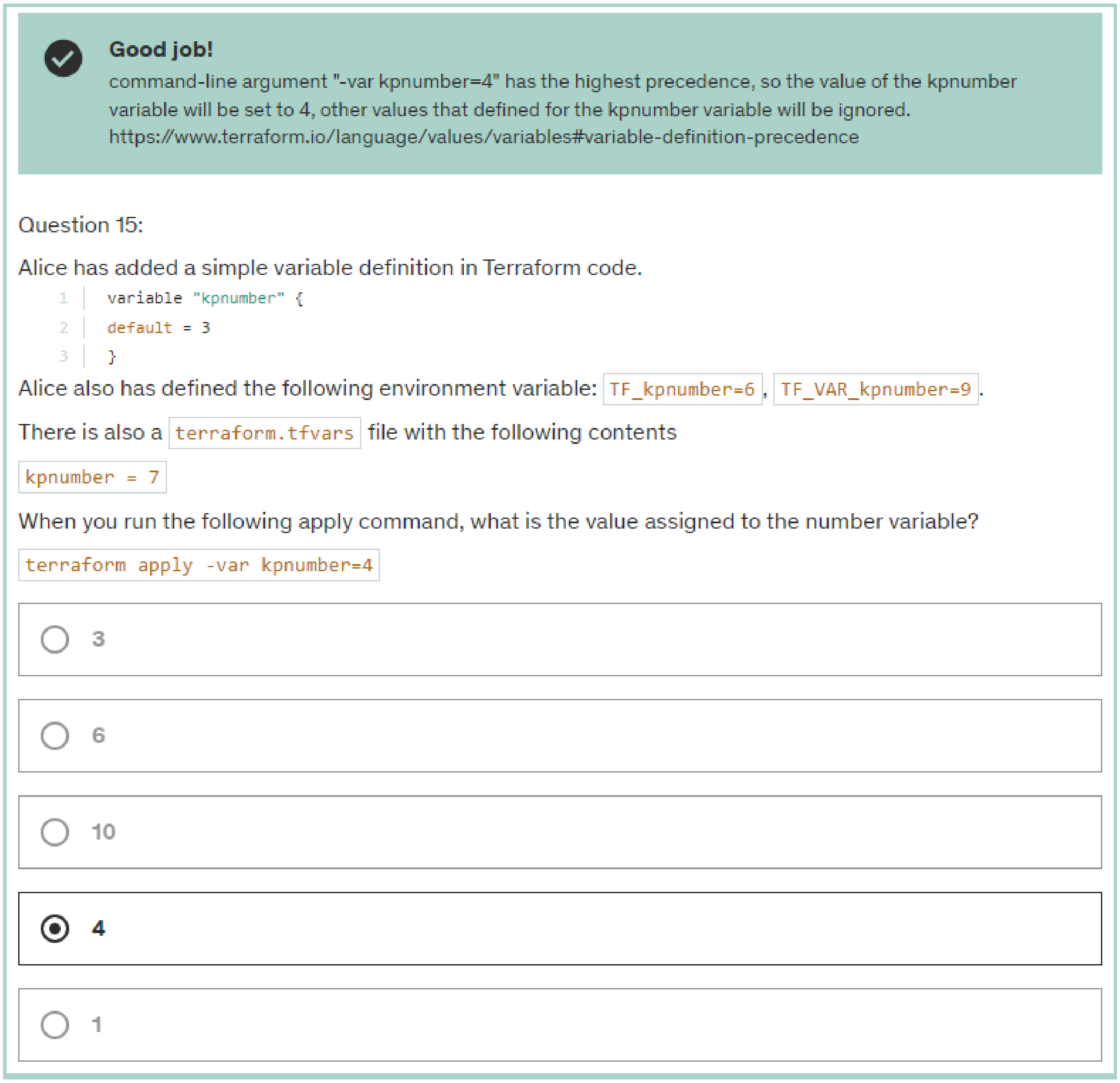
○ dynamic backend

○ terraform import

● **dynamic block**

○ splat expression

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



✓ **Good job!**

command-line argument "-var kpnumber=4" has the highest precedence, so the value of the kpnumber variable will be set to 4, other values that defined for the kpnumber variable will be ignored. https://www.terraform.io/language/values/variables#variable-definition-precedence

Question 15:

Alice has added a simple variable definition in Terraform code.

```
variable "kpnumber" {
default = 3
}
```

Alice also has defined the following environment variable: `TF_kpnumber=6`, `TF_VAR_kpnumber=9`.

There is also a `terraform.tfvars` file with the following contents

`kpnumber = 7`

When you run the following apply command, what is the value assigned to the number variable?

`terraform apply -var kpnumber=4`

○ 3

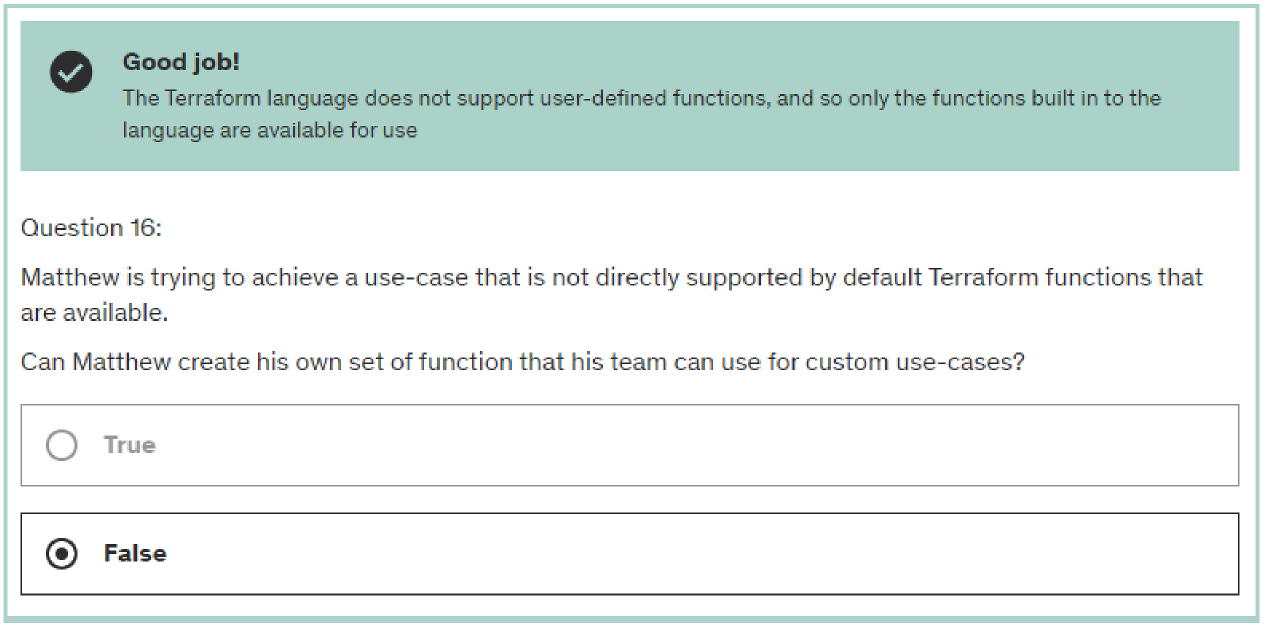
○ 6

○ 10

● 4

○ 1

https://www.terraform.io/language/values/variables#variable-definition-precedence



✓ **Good job!**

The Terraform language does not support user-defined functions, and so only the functions built in to the language are available for use

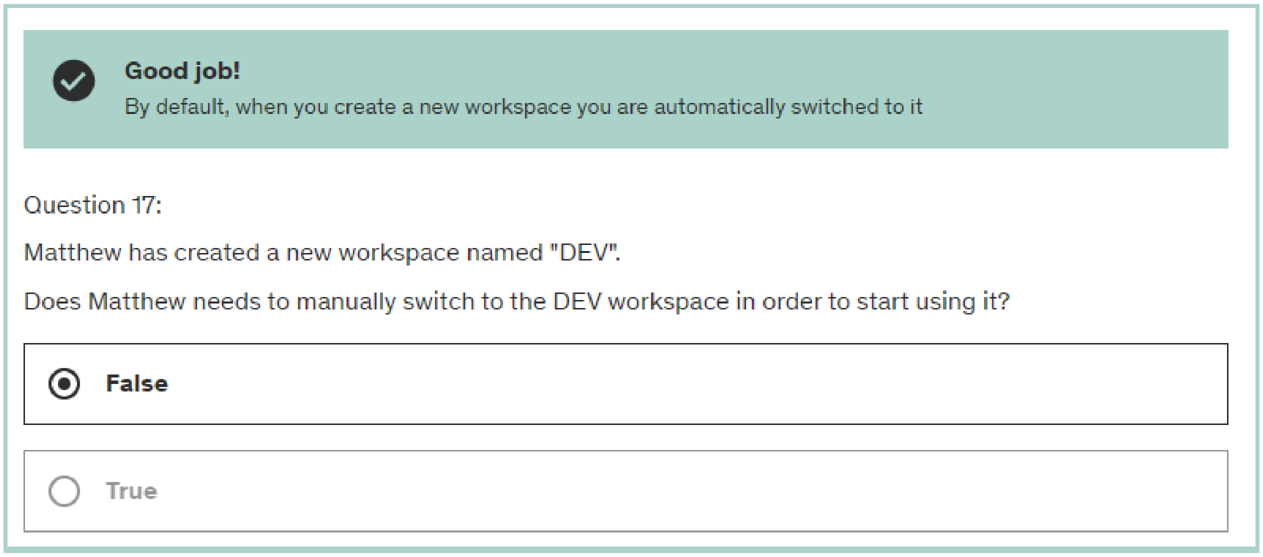
Question 16:

Matthew is trying to achieve a use-case that is not directly supported by default Terraform functions that are available.

Can Matthew create his own set of function that his team can use for custom use-cases?

○ True

● **False**



✓ **Good job!**

By default, when you create a new workspace you are automatically switched to it

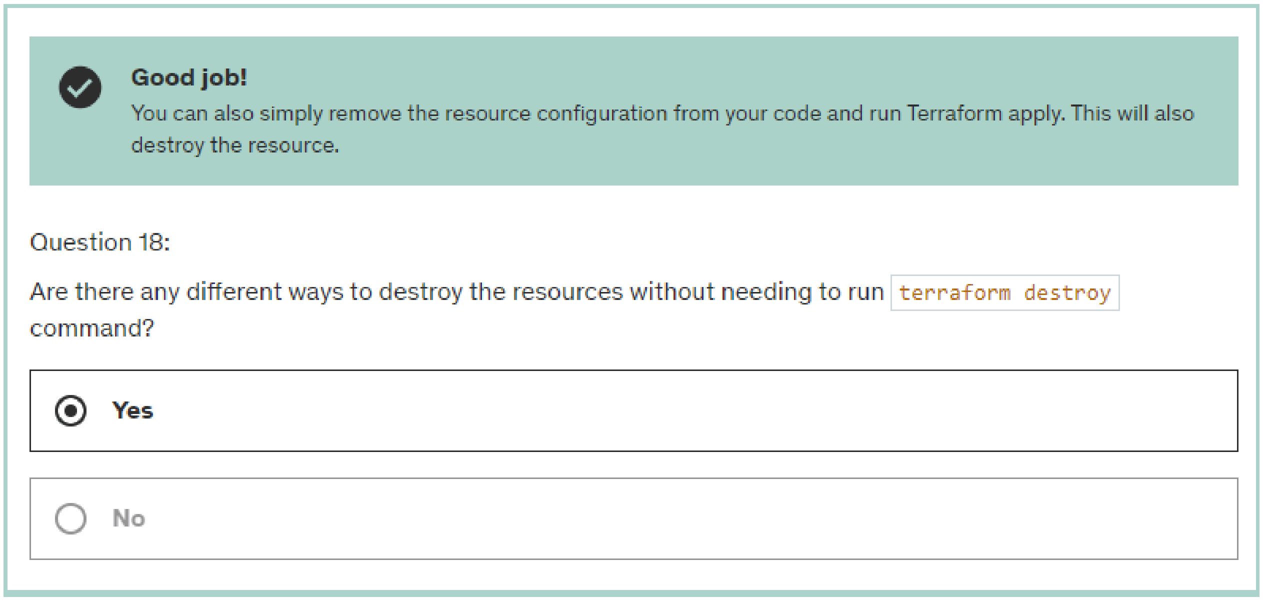
Question 17:

Matthew has created a new workspace named "DEV".

Does Matthew needs to manually switch to the DEV workspace in order to start using it?

● **False**

○ True



✓ **Good job!**

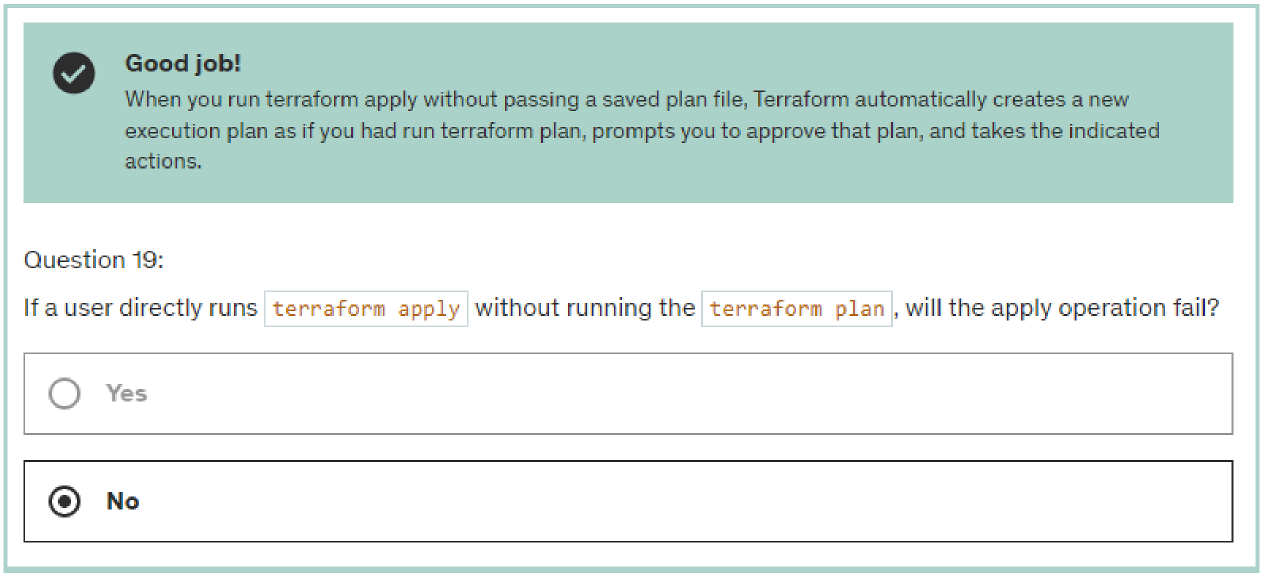
You can also simply remove the resource configuration from your code and run Terraform apply. This will also destroy the resource.

Question 18:

Are there any different ways to destroy the resources without needing to run `terraform destroy` command?

● **Yes**

○ No



✓ **Good job!**

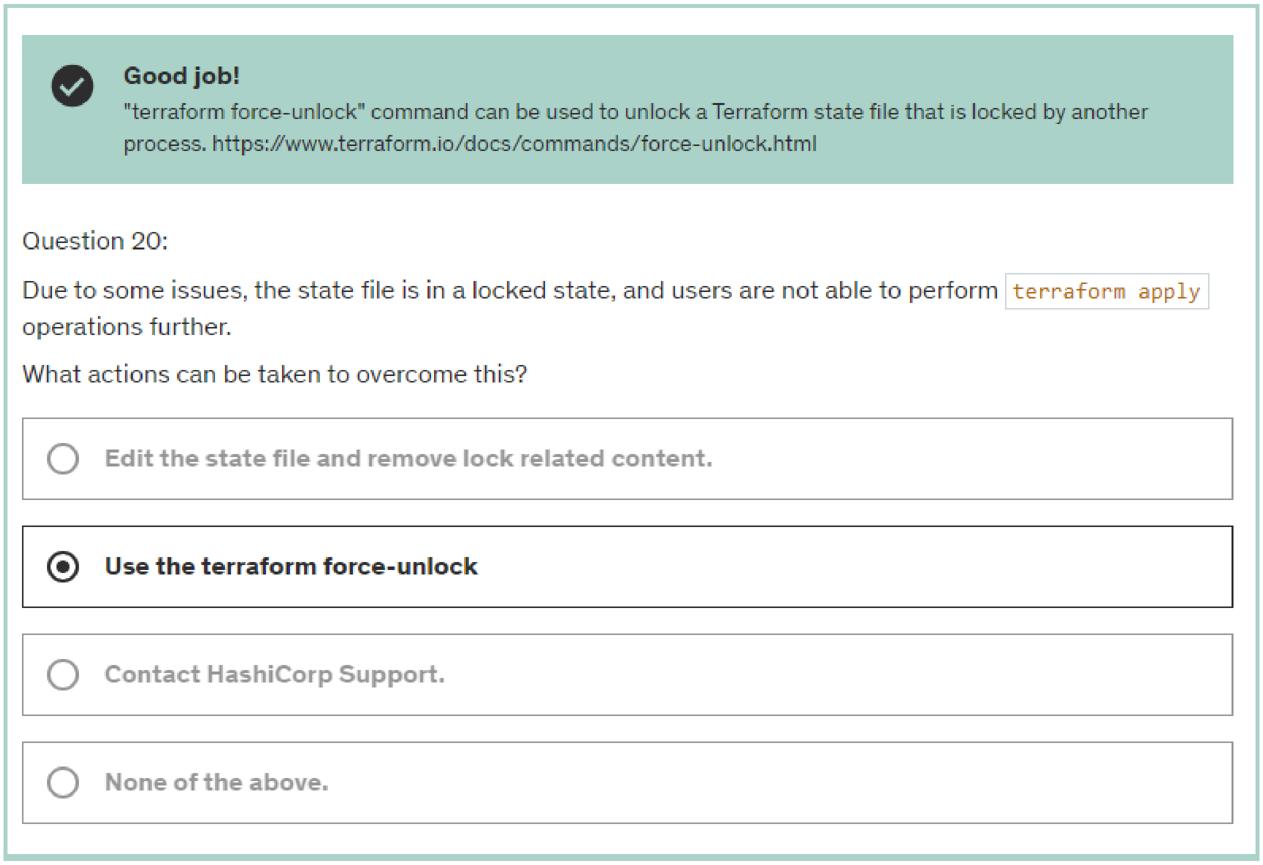
When you run terraform apply without passing a saved plan file, Terraform automatically creates a new execution plan as if you had run terraform plan, prompts you to approve that plan, and takes the indicated actions.

Question 19:

If a user directly runs `terraform apply` without running the `terraform plan`, will the apply operation fail?

○ Yes

● **No**



Good job!
"terraform force-unlock" command can be used to unlock a Terraform state file that is locked by another process. https://www.terraform.io/docs/commands/force-unlock.html

Question 20:

Due to some issues, the state file is in a locked state, and users are not able to perform `terraform apply` operations further.

What actions can be taken to overcome this?

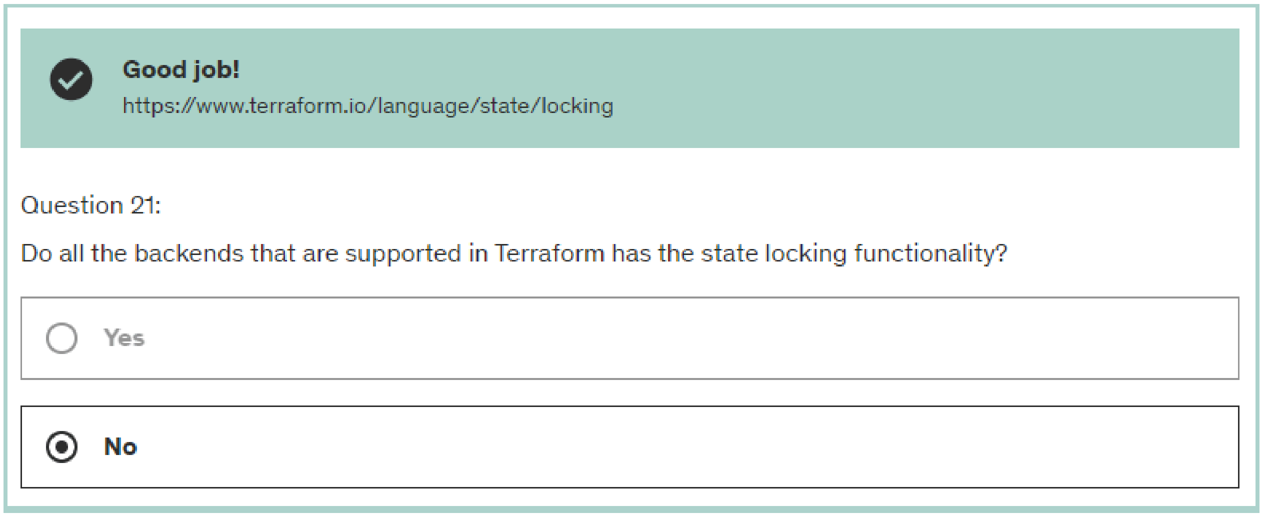
○ Edit the state file and remove lock related content.

● Use the terraform force-unlock

○ Contact HashiCorp Support.

○ None of the above.

https://www.terraform.io/docs/commands/force-unlock.html



Good job!
https://www.terraform.io/language/state/locking

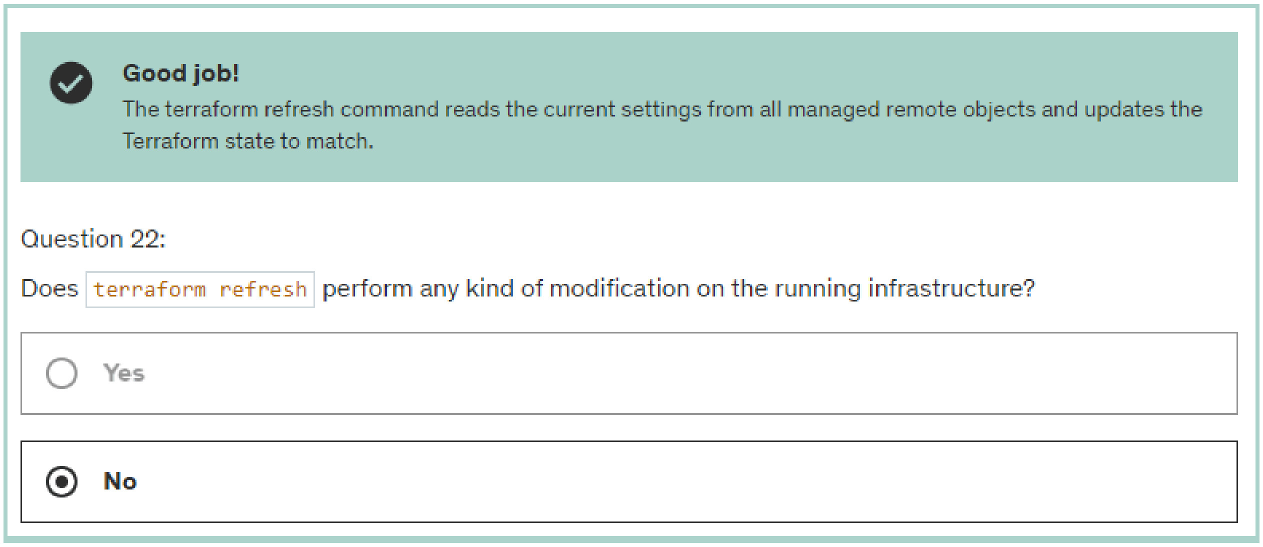
Question 21:

Do all the backends that are supported in Terraform has the state locking functionality?

○ Yes

● No

https://www.terraform.io/language/state/locking



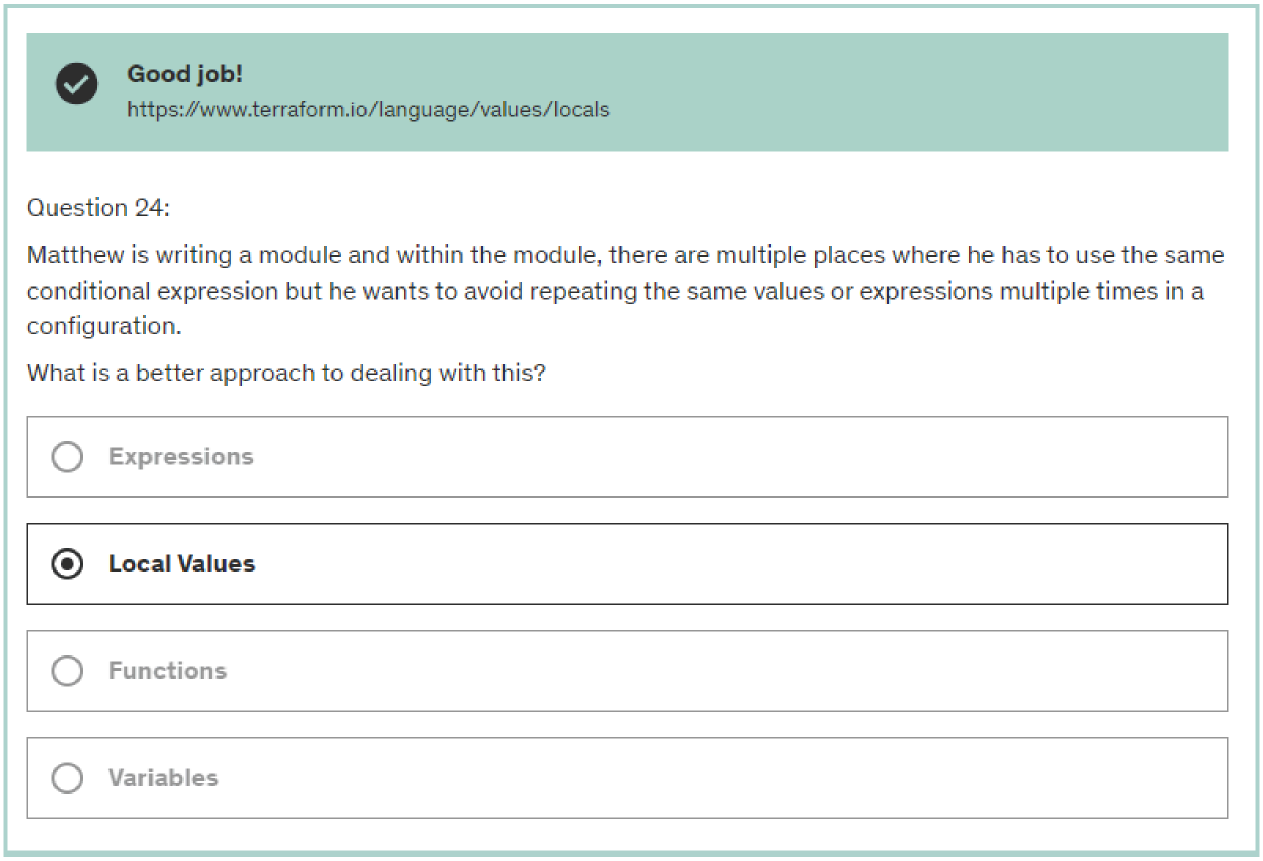
Good job!
The terraform refresh command reads the current settings from all managed remote objects and updates the Terraform state to match.

Question 22:

Does `terraform refresh` perform any kind of modification on the running infrastructure?

○ Yes

◉ **No**



Good job!
https://www.terraform.io/language/values/locals

Question 24:

Matthew is writing a module and within the module, there are multiple places where he has to use the same conditional expression but he wants to avoid repeating the same values or expressions multiple times in a configuration.

What is a better approach to dealing with this?

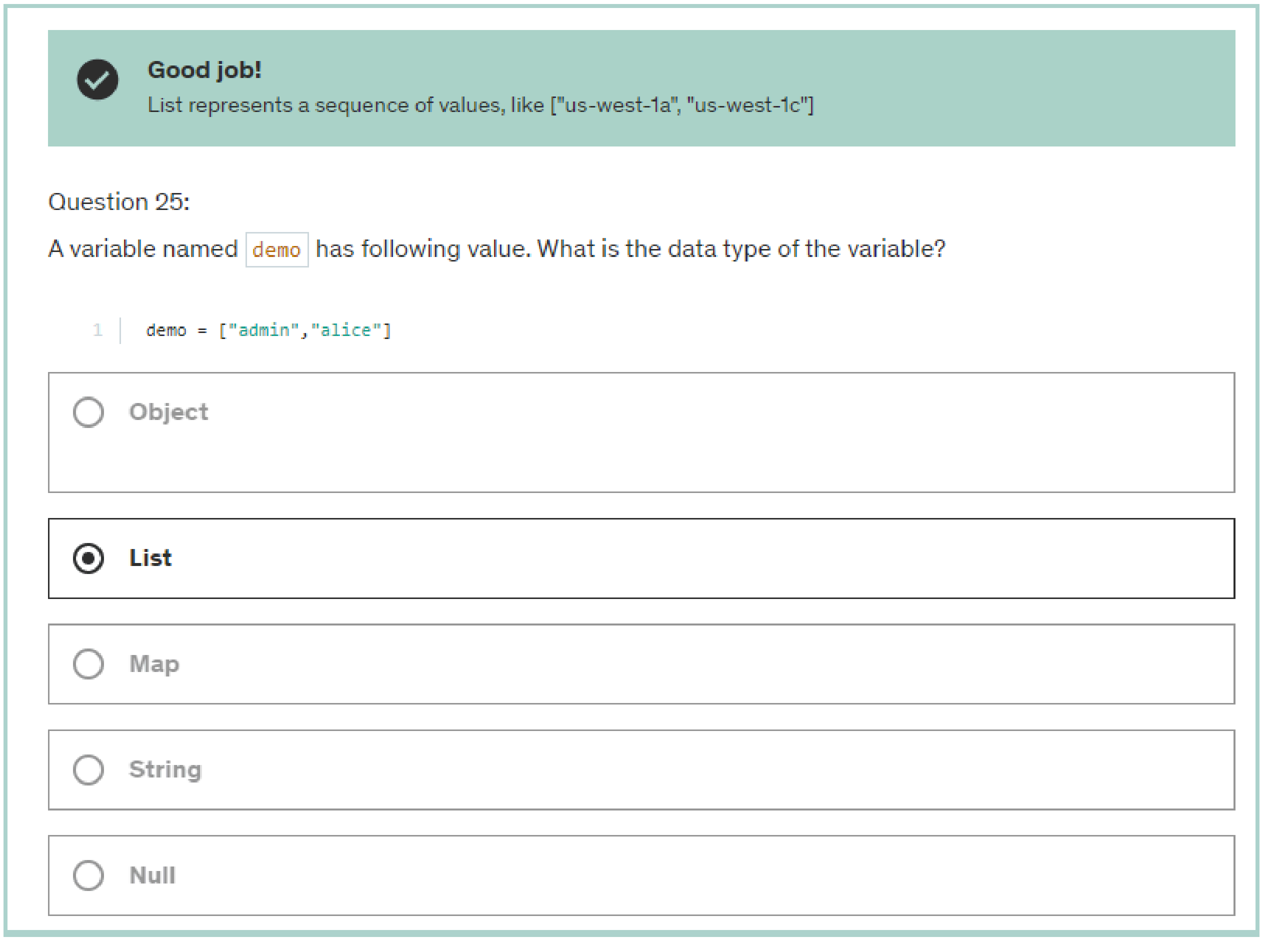
○ Expressions

◉ **Local Values**

○ Functions

○ Variables

https://www.terraform.io/language/values/locals



Good job!

List represents a sequence of values, like ["us-west-1a", "us-west-1c"]

Question 25:

A variable named `demo` has following value. What is the data type of the variable?

```
demo = ["admin","alice"]
```

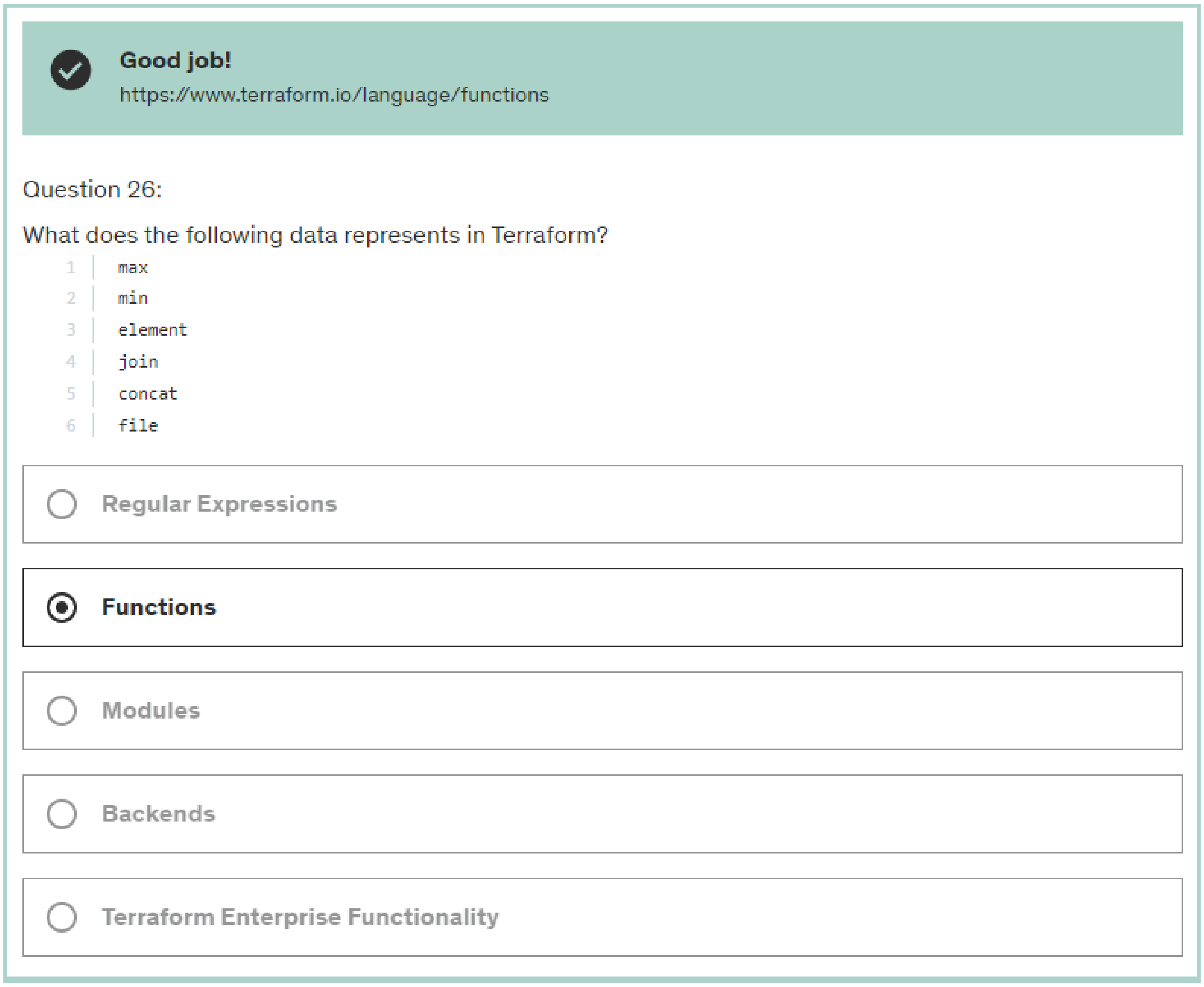
○ Object

● **List**

○ Map

○ String

○ Null



✓ **Good job!**

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

Question 26:

What does the following data represents in Terraform?

```
max
min
element
join
concat
file
```

○ Regular Expressions

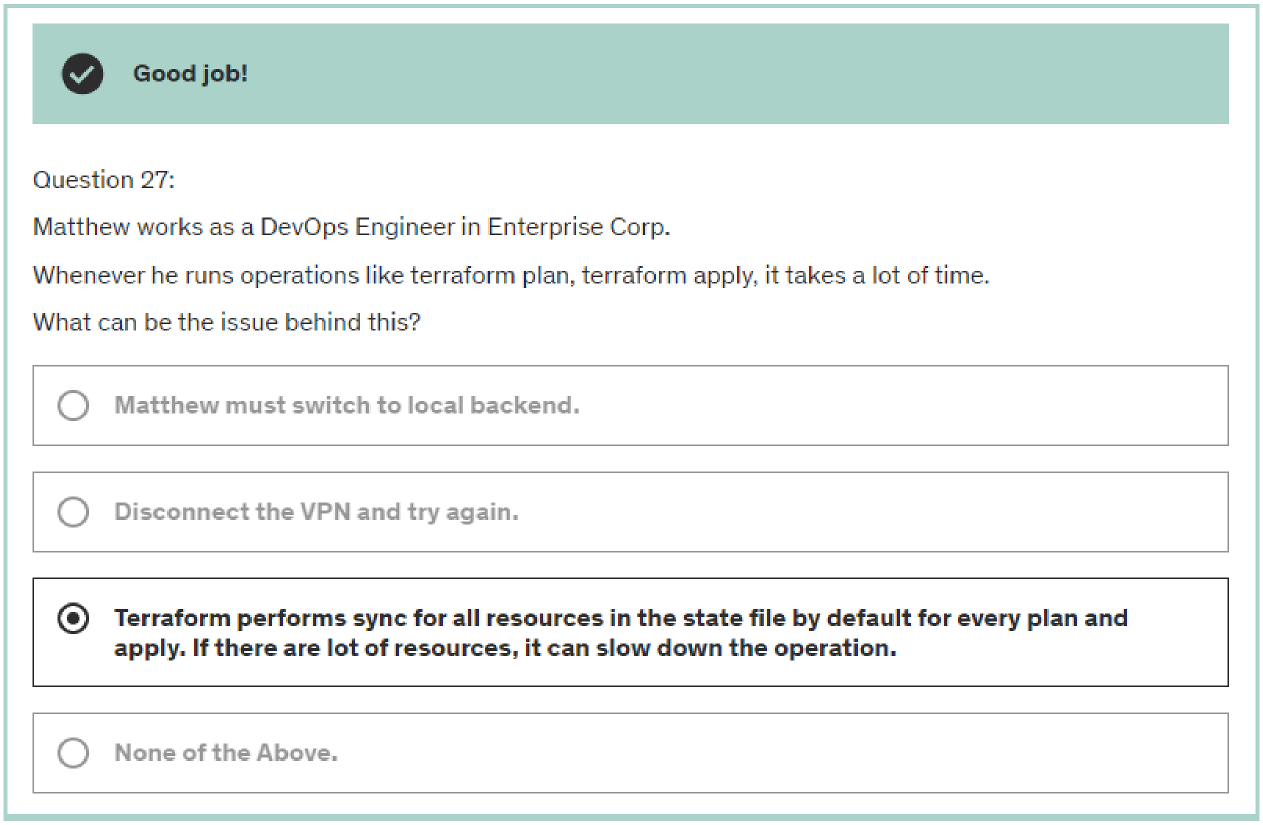
● **Functions**

○ Modules

○ Backends

○ Terraform Enterprise Functionality

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



✓ **Good job!**

Question 27:

Matthew works as a DevOps Engineer in Enterprise Corp.

Whenever he runs operations like terraform plan, terraform apply, it takes a lot of time.

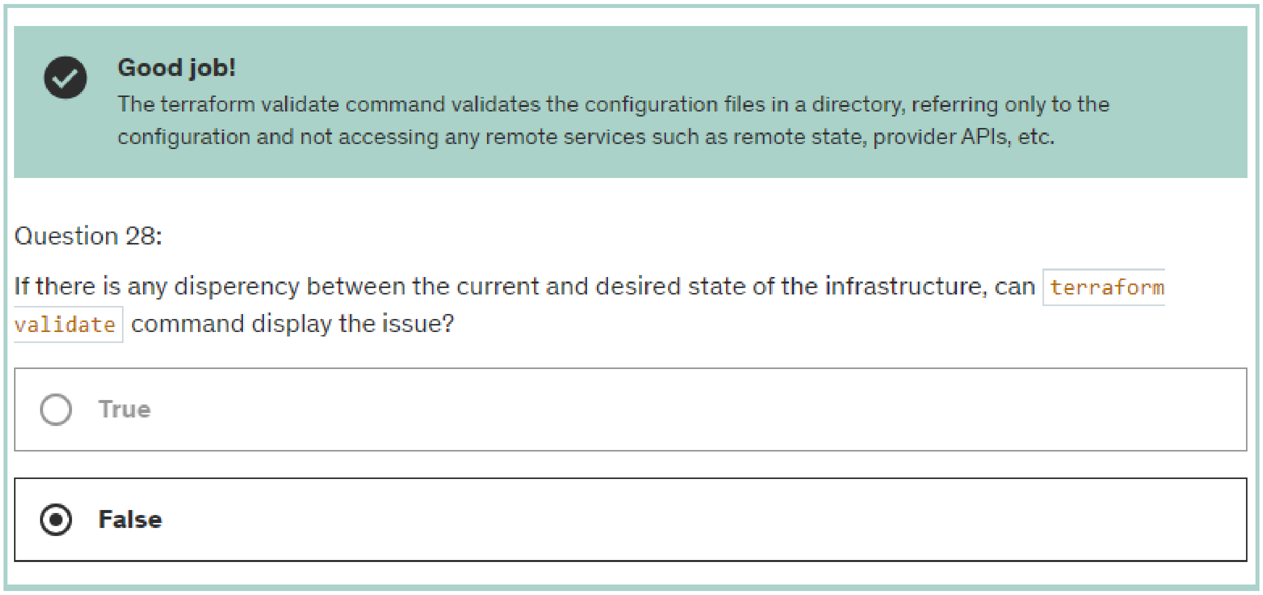
What can be the issue behind this?

○ Matthew must switch to local backend.

○ Disconnect the VPN and try again.

● **Terraform performs sync for all resources in the state file by default for every plan and apply. If there are lot of resources, it can slow down the operation.**

○ None of the Above.



✓ **Good job!**

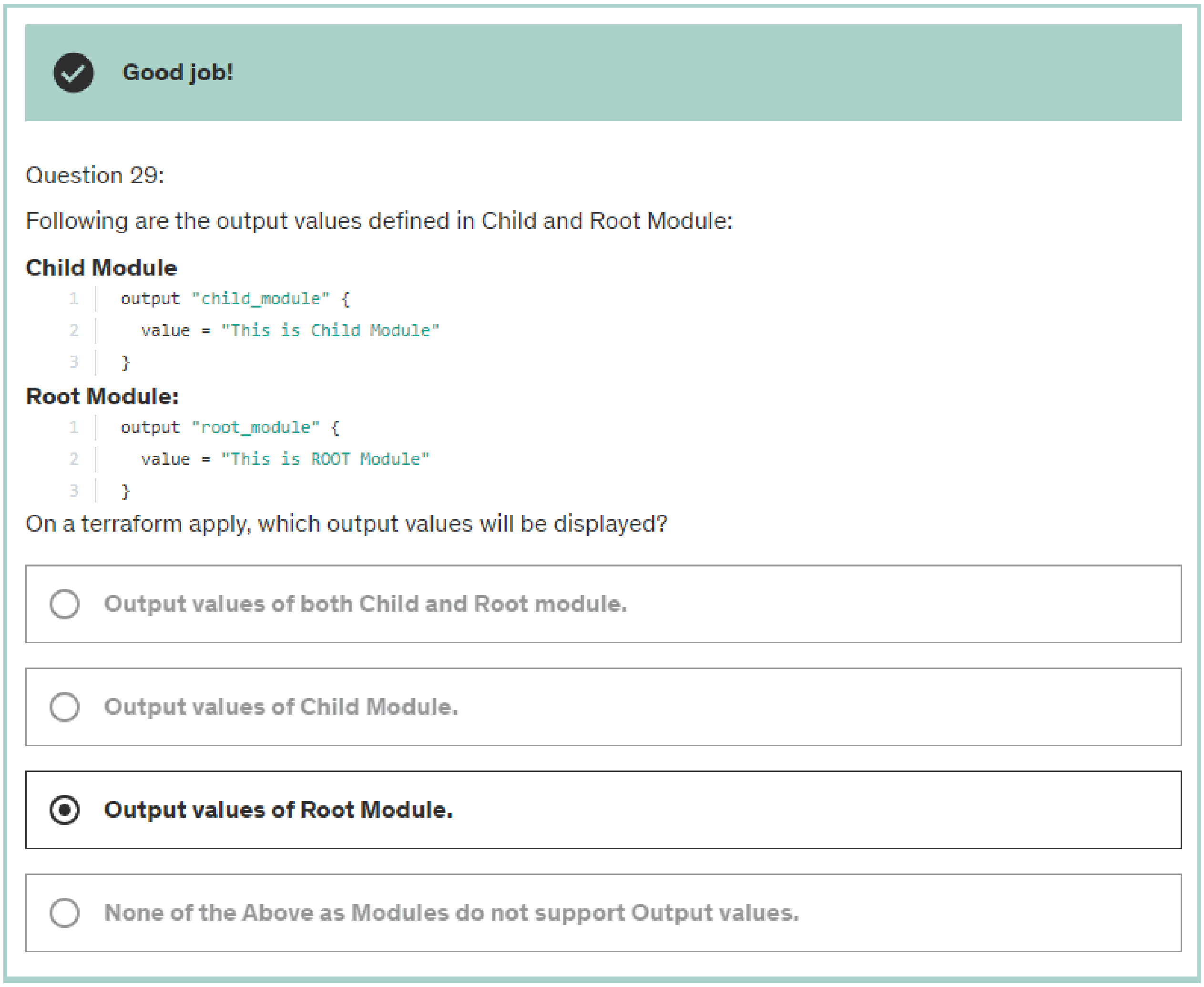
The terraform validate command validates the configuration files in a directory, referring only to the configuration and not accessing any remote services such as remote state, provider APIs, etc.

Question 28:

If there is any disperency between the current and desired state of the infrastructure, can `terraform validate` command display the issue?

○ True

● **False**



✓ **Good job!**

Question 29:

Following are the output values defined in Child and Root Module:

Child Module

```
output "child_module" {
  value = "This is Child Module"
}
```

Root Module:

```
output "root_module" {
  value = "This is ROOT Module"
}
```

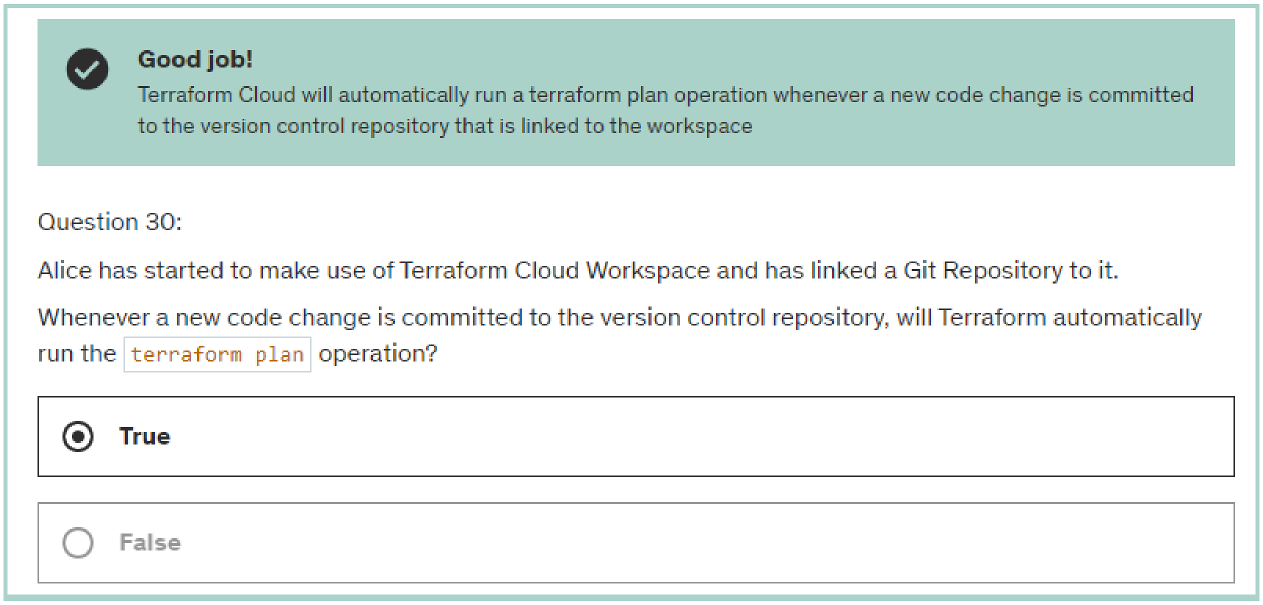
On a terraform apply, which output values will be displayed?

○ Output values of both Child and Root module.

○ Output values of Child Module.

● **Output values of Root Module.**

○ None of the Above as Modules do not support Output values.



✓ **Good job!**
Terraform Cloud will automatically run a terraform plan operation whenever a new code change is committed to the version control repository that is linked to the workspace

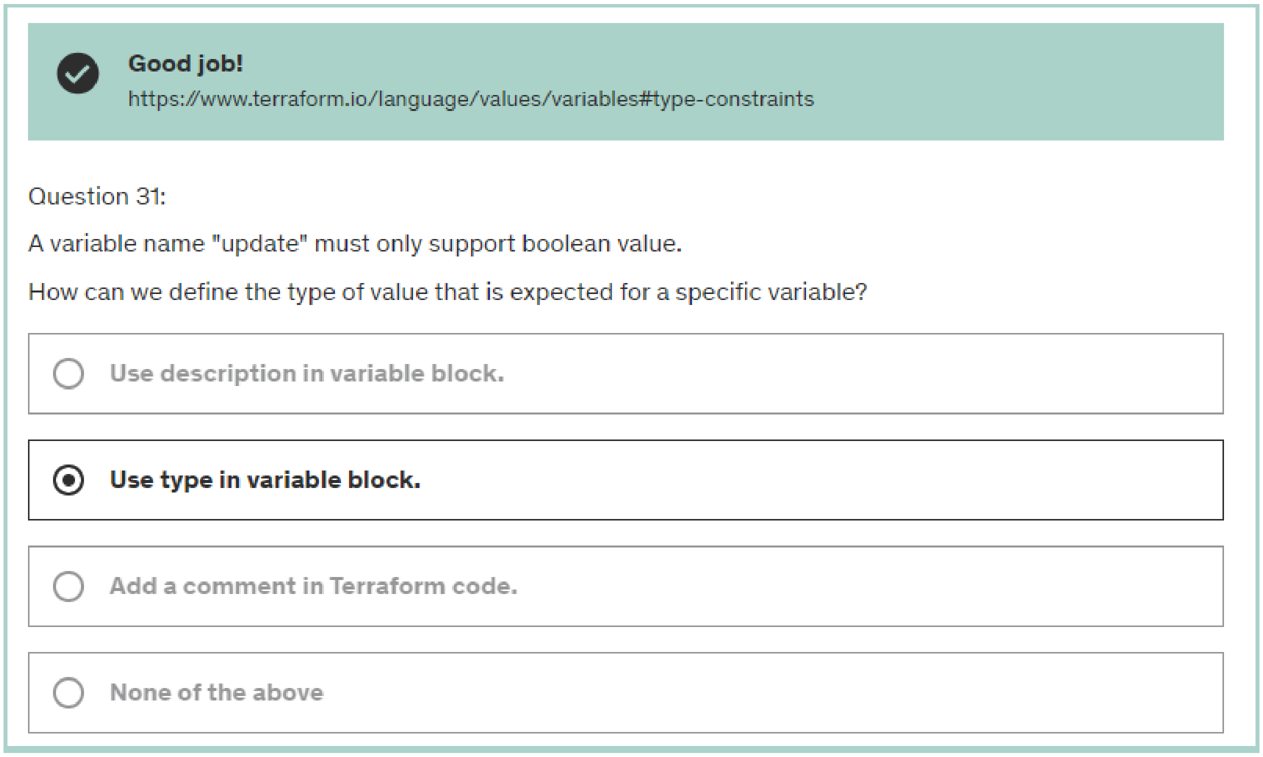
Question 30:

Alice has started to make use of Terraform Cloud Workspace and has linked a Git Repository to it.

Whenever a new code change is committed to the version control repository, will Terraform automatically run the `terraform plan` operation?

● **True**

○ False



Good job!
https://www.terraform.io/language/values/variables#type-constraints

Question 31:

A variable name "update" must only support boolean value.

How can we define the type of value that is expected for a specific variable?

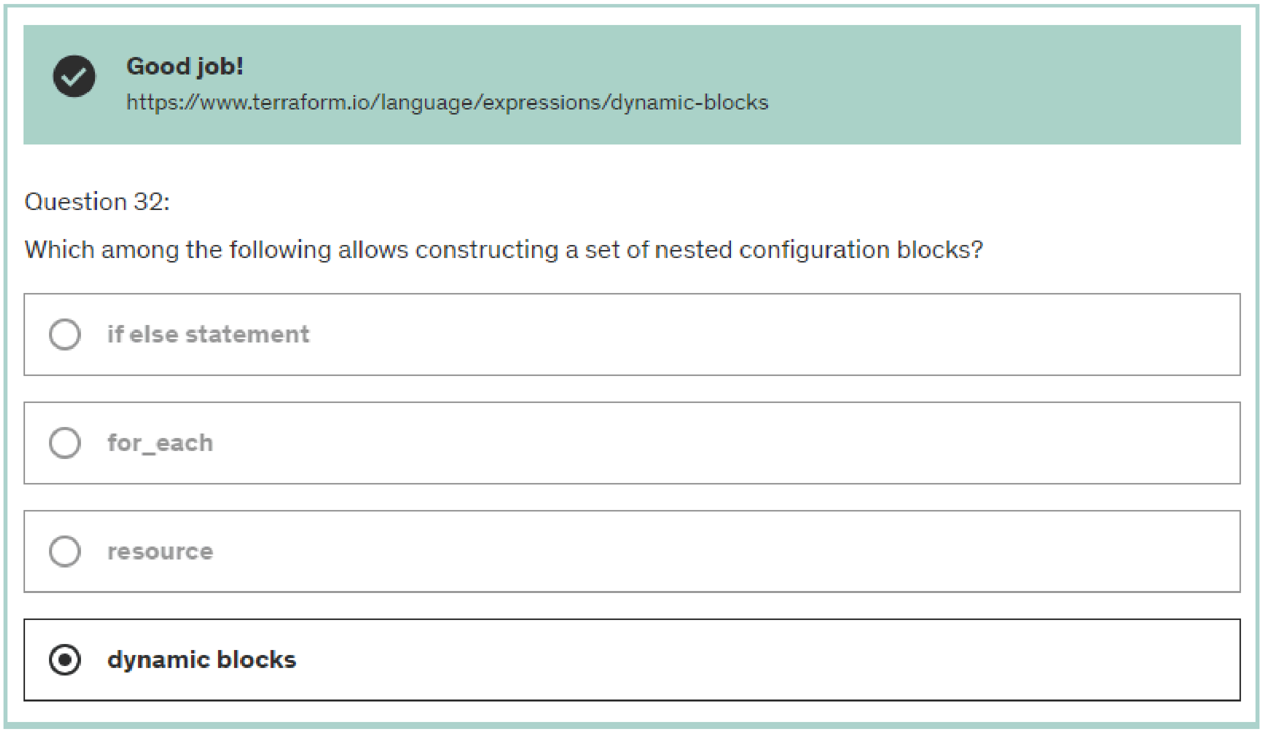
○ Use description in variable block.

● **Use type in variable block.**

○ Add a comment in Terraform code.

○ None of the above

https://www.terraform.io/language/values/variables#type-constraints



Good job!
https://www.terraform.io/language/expressions/dynamic-blocks

Question 32:

Which among the following allows constructing a set of nested configuration blocks?

○ if else statement

○ for_each

○ resource

● **dynamic blocks**

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