



Terraform: Loops in TerraFrom HCL

Terraform : Deployment Automation

- Terraform offers several different looping constructs.
 - **Count** : Loop Over resources
 - **For** : Loop over lists and Maps
 - **For_each** : Loop over resources and inline blocks within a resource.

Terraform : Deployment Automation

- **count parameter** - count is used to loop over the resources. This can be used to create the multiple resources in TF.
- Create single IAM User -

```
provider "aws" {  
  region = "us-east-2"  
}  
  
resource "aws_iam_user" "user_example" {  
  name = "myuser"  
}
```

- What if we want to Create Multiple Users?

```
for (i = 0; i < 5; i++) {  
  resource "aws_iam_user" "user_example" {  
    name = "myuser"  
  }  
}
```

Terraform : Deployment Automation

- Terraform doesn't have loops on resources like Programming languages.
- Count can be used to create multiple copies of resource in TF.

```
provider "aws" {  
  region = "us-east-2"  
}  
  
resource "aws_iam_user" "user_example" {  
  count = 3  
  name = "myuser.${count.index}"  
}
```

Terraform : Deployment Automation

➤ **for parameter** - for is used to iterate over Lists and Maps. For loop is used to generate single Value.

➤ Syntax for 'for' loop -

[for <ITEM> in <LIST> : <OUTPUT>]

```
variable "names" {
  description = "A list of names"
  type       = list(string)
  default    = ["mark", "trinity", "john"]
}

output "upper_names" {
  value = [for name in var.names : upper(name)]
}
```

Terraform : Deployment Automation

► Syntax map -

[for <KEY>, <VALUE> in <MAP> : <OUTPUT>]

```
variable "program" {
  description = "map"
  type       = map(string)
  default    = {
    mark    = "software engineer"
    trinity = "AI Program"
    john    = "machine operator"
  }
}

output "roles" {
  value = [for name, role in var.program : "${name} is the - ${role}"]
}
```

Terraform : Deployment Automation

- **for_each parameter** - for_each is used to create multiple copies of resource or inline blocks.
- Syntax for 'for_each' loop -

```
resource "<PROVIDER>_<TYPE>" "<NAME>" {  
  for_each = <COLLECTION>  
  [CONFIG ...]  
}
```

```
variable "user_names" {  
  description = "Create IAM users with these names"  
  type        = list(string)  
  default     = ["mark", "trinity", "john"]  
}  
  
resource "aws_iam_user" "user_example" {  
  for_each = toset(var.user_names)  
  name     = each.value  
}
```

Will see you in Next Lecture...

Thank you!

A close-up photograph of a hand holding a black marker, writing the words 'Thank you!' in a cursive script on a white surface. The hand is positioned on the right side of the frame, with the fingers gripping the marker. The text is written in a dark, fluid cursive style.

See you in next lecture ...