Terraform

Set

Creation of 5 users

1. Main.tf

```
provider "aws" {
   region= var.region1
   access_key = var.access_key
   secret_key = var.secret_key
}
resource "aws_iam_user" "iamusers" {
  for_each = var.usernames
  name = each.value
  }
```

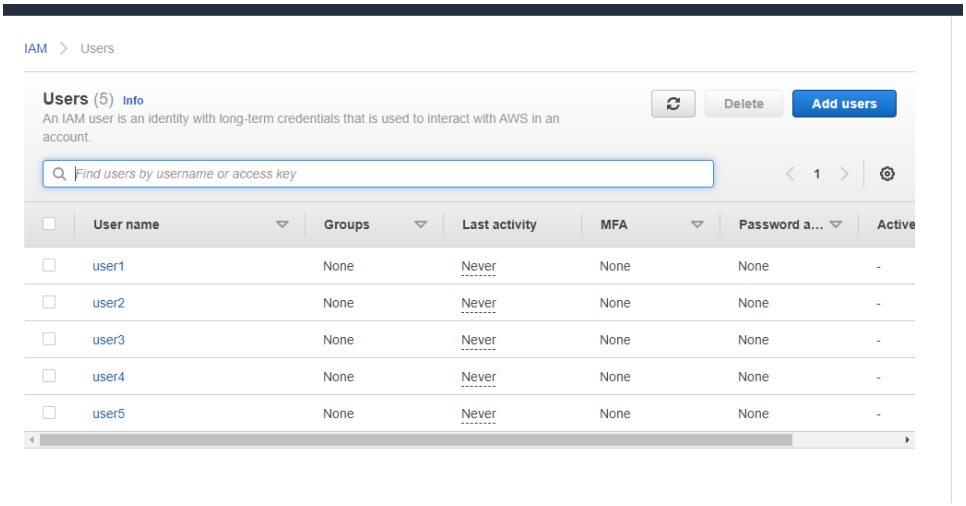
Variable.tf

```
variable "access_key" {
   type = string
}
variable "secret_key" {
   type = string
}
variable "region1" {
   default = "us-east-1"
}
variable "region2" {
   default = "us-west-2"
}
variable "usernames"{
   description = "iam_user"
```

```
  type = set(string)
  default = ["user1", "user2", "user3", "user4", "user5"]
}
}
```

Terraform.tfvars

```
access_key = "AKIA6LFFGWBXSELWSXC"
secret_key = "kxGntAwaCUC0bXqVNV9+RVNa2hKaF7uB4DCACMy"
```



Tuple

Creation of ec2 using Tuple variable

Main.tf

```
provider "aws" {
   region= var.region1
   access_key = var.access_key
   secret_key = var.secret_key
}

resource "aws_instance" "ec2_tuple"{
  ami = var.instance_details[0]
  instance_type = var.instance_details[1]
  count = var.instance_details[2]
  associate_public_ip_address = var.instance_details[3]
  tags = {
    "Name" = "tuple-ec2"
  }
}
```

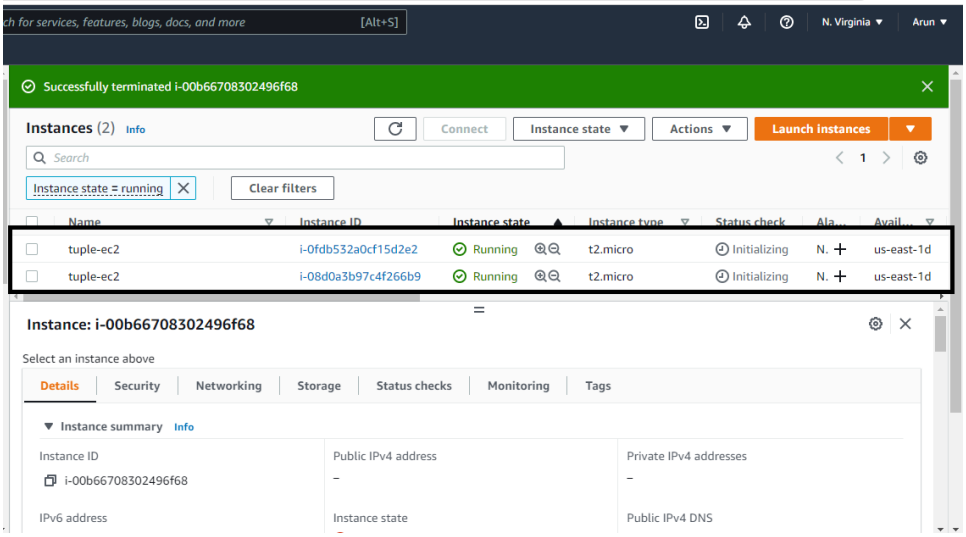
Variable.tf

```
variable "access_key" {
   type = string
}
variable "secret_key" {
   type = string
}
variable "region1" {
   default = "us-east-1"
}
variable "region2" {
   default = "us-west-2"
}
variable "instance_details" {
   description = "provides instance details"
   type = tuple([string, string,number, bool])
   default = (["ami-0c02fb55956c7d316","t2.micro","2","true"])
}
```

Terraform.tfvars

```
access_key = "AKIA6LFFBWBXSELWSXC"
secret_key = "kxGntAwCU9C0bXqVNV9+RVNa2hKaF7uB4DCACMy"
```

terraform apply -auto-approve -var-file terraform.tfvars



Object
Creation of ec2 using object

Main.tf

```
provider "aws" {
   region= var.region1
   access_key = var.access_key
   secret_key = var.secret_key
}
resource "aws_instance" "ec2_object"{
 ami = var.instance_details_object.ami
 instance_type = var.instance_details_object.instance_type
 count = var.instance_details_object.no_of_instances
 associate_public_ip_address = var.instance_details_object.Public_ip
 tags = {
  "Name" = "object-ec2"
 }
}
```

Variable.tf

```
variable "access_key" {
   type = string
}
variable "secret_key" {
   type = string
}
variable "region1" {
   default = "us-east-1"
}
variable "region2" {
   default = "us-west-2"
```

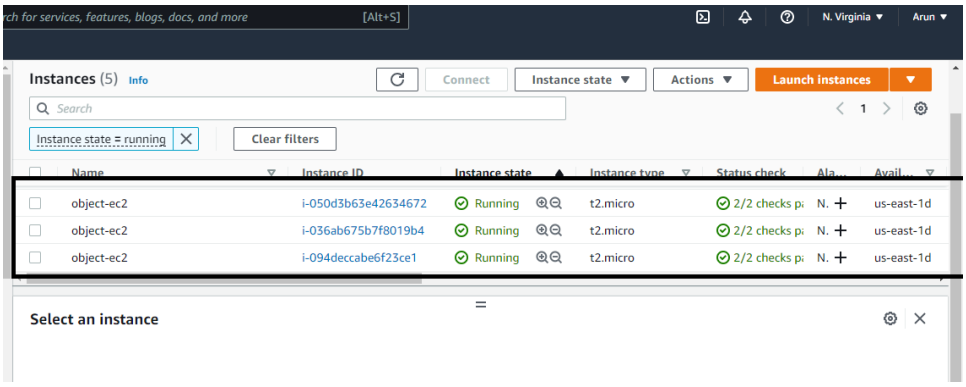
```
}
variable "usernames"{
   description = "iam_user"
   type = set(string)
   default = ["user1", "user2", "user3", "user4", "user5"]
}

variable "instance_details_object" {
   description = "provides instance details"
   type = object({ami=string, instance_type=string,no_of_instances=number, Public_ip=bool})
   default = ({ami="ami-0c02fb55956c7d316", instance_type="t2.micro",no_of_instances="3", Public_ip="true"})
}
```

Terraform.tfvars

```
access_key = "AKIA6LFGBWBXSELWSXC"
secret_key = "kxGntAaCU9C0bXqVNV9+RVNa2hKaF7uB4DCACMy"
```

terraform apply -auto-approve -var-file terraform.tfvars



For Loop and Output

Print the list of Iam_user

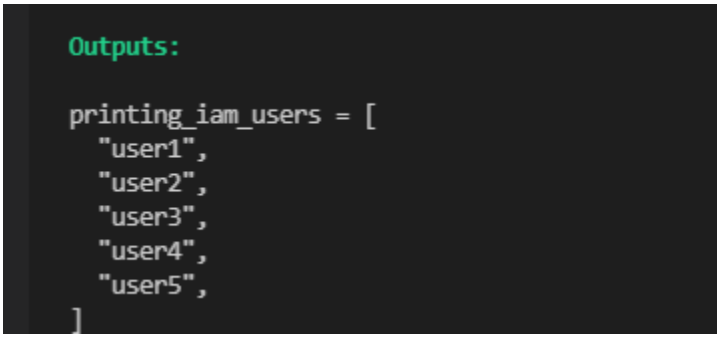
Main.tf

```
provider "aws" {
   region= var.region1
   access_key = var.access_key
   secret_key = var.secret_key
}
resource "aws_iam_user" "iamusers" {
  for_each = var.usernames
  name = each.value
}
output "printing_iam_users" {
  value = [for name in var.usernames : name]
}
```

Variables.tf

```
variable "access_key" {
   type = string
}
variable "secret_key" {
   type = string
}
variable "region1" {
   default = "us-east-1"
}
variable "region2" {
```

```
  default = "us-west-2"
}
variable "usernames"{
  description = "iam_user"
  type = set(string)
  default = ["user1", "user2", "user3", "user4", "user5"]
}
```



Any variable

We can also use any in place of variable types

```
variable "ami_id" {
  type = any
  default =  “ami-0c02fb55956c7d316”
}
```