

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~  
$ cd https://github.com/NITINSRIGANESH/terraform-ec2-multi-region  
bash: cd: https://github.com/NITINSRIGANESH/terraform-ec2-multi-region: No such  
file or directory
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~  
$ git clone https://github.com/NITINSRIGANESH/terraform-ec2-multi-region  
Cloning into 'terraform-ec2-multi-region'...  
remote: Enumerating objects: 3, done.  
remote: Counting objects: 100% (3/3), done.  
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)  
Receiving objects: 100% (3/3), done.
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~  
$ cd terraform-ec2-multi-region
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ terraform -v  
Terraform v1.10.3  
on windows_386
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ touch main.tf
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ touch variables.tf
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ touch outputs.tf
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ terraform init  
Initializing the backend...  
Initializing provider plugins...
```


**Terraform has been successfully initialized!**

You may now begin working with Terraform. Try running "terraform plan" to see any changes that are required for your infrastructure. All Terraform commands should now work.

If you ever set or change modules or backend configuration for Terraform, rerun this command to reinitialize your working directory. If you forget, other commands will detect it and remind you to do so if necessary.

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$
```

Files

 main

🔍 Go to file


 README.md


main.tf

output.tf

 variables.tf

[terraform-ec2-multi-region / main.tf](#) 


 **NITINSRIGANESH** Update main.tf

7b9a8a2 · 7 minutes ago  **History**

Code

## Blame

6 lines (5 loc) · 179 Bytes

 Code 55% faster with GitHub Copilot

Raw



7



9



```
1 resource "aws_instance" "example_us_east_1" {
2     provider = aws.us-east-1
3     ami      = "ami-0c02fb55956c7d316" # Update AMI ID or other parameters
4     instance_type = "t2.micro"
5 }
```

## Files


main





Q Go to file


 README.md

output.tf

 variables.tf


[terraform-ec2-multi-region / output.tf](#) 

 **NITINSRIGANESH** Create output.tf

 History

Code

6 lines (6 loc) · 133 Bytes

 Code 55% faster with GitHub Copilot

Raw



```
1 output "instance_ips" {
2     value = [
3         aws_instance.example_us_east_1.public_ip,
4         aws_instance.example_us_west_1.public_ip
5     ]
6 }
```

**Files**

main + 🔍

Go to file t

- README.md
- main.tf
- output.tf
- variables.tf**

terraform-ec2-multi-region / variables.tf

NITINSRIGANESH Create variables.tf d0a11f1 · 12 minutes ago History

Code Blame 7 lines (6 loc) · 95 Bytes Code 55% faster with GitHub Copilot Raw Copy Download Edit

```
1 variable "region1" {
2     default = "us-east-1"
3 }
4
5 variable "region2" {
6     default = "us-west-1"
7 }
```



```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~  
$ cd https://github.com/NITINSRIGANESH/terraform-ec2-multi-region  
bash: cd: https://github.com/NITINSRIGANESH/terraform-ec2-multi-region: No such  
file or directory
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~  
$ git clone https://github.com/NITINSRIGANESH/terraform-ec2-multi-region  
Cloning into 'terraform-ec2-multi-region'...  
remote: Enumerating objects: 3, done.  
remote: Counting objects: 100% (3/3), done.  
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)  
Receiving objects: 100% (3/3), done.
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~  
$ cd terraform-ec2-multi-region
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ terraform -v  
Terraform v1.10.3  
on windows_386
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ touch main.tf
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ touch variables.tf
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ touch outputs.tf
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$ terraform init  
Initializing the backend...  
Initializing provider plugins...
```

**Terraform has been successfully initialized!**

You may now begin working with Terraform. Try running "terraform plan" to see any changes that are required for your infrastructure. All Terraform commands should now work.

If you ever set or change modules or backend configuration for Terraform, rerun this command to reinitialize your working directory. If you forget, other commands will detect it and remind you to do so if necessary.

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region (main)  
$
```

**No changes.** Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region/terraform-ec2-multi-region (main)
$ terraform apply
```

**No changes.** Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.

**Apply complete! Resources: 0 added, 0 changed, 0 destroyed.**

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region/terraform-ec2-multi-region (main)
$ ^C
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region/terraform-ec2-multi-region (main)
$ terraform plan
```

**No changes.** Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region/terraform-ec2-multi-region (main)
$ terraform apply
```

**No changes.** Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.

**Apply complete! Resources: 0 added, 0 changed, 0 destroyed.**

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region/terraform-ec2-multi-region (main)
$ ^C
```

```
Nitin.S@DESKTOP-HF82NIK MINGW64 ~/terraform-ec2-multi-region/terraform-ec2-multi-region (main)
$
```