

< Hide

Module 9 - Kubernetes II	▼
Module 10 - Terraform	^
Terraform PPT	
Assignment 1 - Terraform	
Assignment 2 - Terraform	
Assignment 3 - Terraform	
Assignment 4 - Terraform	
Assignment 5 - Terraform	
Case Study - Terraform	
ELK	▼

DevOps Certification Training

IntelliPaat

Terraform Assignment - 1

You have been asked to:

- Create an EC2 service in the default subnet in the ohio region

Download Attachment

```
root@ip-172-31-93-95:~# mkdir terraform
root@ip-172-31-93-95:~# ls
snap terraform
root@ip-172-31-93-95:~# cd terraform/
root@ip-172-31-93-95:~/terraform#
root@ip-172-31-93-95:~/terraform# ls
root@ip-172-31-93-95:~/terraform#
root@ip-172-31-93-95:~/terraform# vi creds.tf
root@ip-172-31-93-95:~/terraform# cat creds.tf
# Configure the AWS Provider
provider "aws" {
  region = "us-east-1"
}
```

```
root@ip-172-31-93-95:~/terraform# ls
creds.tf
root@ip-172-31-93-95:~/terraform# terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v4.8.0...
- Installed hashicorp/aws v4.8.0 (signed by HashiCorp)
```

Terraform has created a lock file `.terraform.lock.hcl` to record the provider selections it made above. Include this file in your version control repository so that Terraform can guarantee to make the same selections by default when you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see

```
root@ip-172-31-93-95:~/terraform# apt install awscli
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  docutils-common fontconfig fontconfig-config fonts-dejavu-core
  fonts-droid-fallback fonts-ghostscript fonts-groff fonts-gsfonts
  hicolor-icon-theme imagemagick imagemagick-6-common imagemagick-6.q16
  libavahi-client3 libavahi-common-data libavahi-common3 libcairo2 libcups2
  libdatatriel libdjvulibre-text libdjvulibre21 libfftw3-double3 libfontconfig1
  libgomp1 libgraphite2-3 libgs9 libgs9-common libharfbuzz0b libice6 libidn11
  libijs-0.35 libilmbase24 libimagequant0 libjbig0 libjbig2dec0 libjpeg-turbo8
  libjpeg8 liblcms2-2 liblqr-1-0 libmagickcore-6.q16-6 libmagickcore-6.q16-6-extra
  libmagickwand-6.q16-6 libnetpbm10 libopenexr24 libopenjp2-7 libpango-1.0-0
  libpangocairo-1.0-0 libpangoft2-1.0-0 libpaper-utils libpaper1 libpixman-1-0
  libsm6 libthai-data libthai0 libtiff5 libwebp6 libwebpdemux2 libwebpmux3
  libwmf0.2-7 libxaw7 libxcb-render0 libxcb-shm0 libxmu6 libxpm4 libxrender1
  libxt6 netpbm poppler-data psutils python3-boto python3-dateutil
  python3-docutils python3-jmespath python3-olefile python3-pil python3-pygments
  python3-roman python3-rsa python3-s3transfer sgml-base x11-common xml-core
Suggested packages:
  fonts-ghostscript fonts-freefont-otf | fonts-freefont-ttf fonts-texgyre ghostscript-x
  imagemagick-doc autotrace cups-bsd | lpr | lprng enscript ffmpeg gimp gnuplot
  grads graphviz hp2xx html2ps libwmf-bin mplayer povray radiance sane-utils
  texlive-base-bin transfig ufraw-batch xdg-utils cups-common libfftw3-bin
  libfftw3-dev liblcms2-utils inkscape libjxr-tools libwmf0.2-7-gtk poppler-utils
  fonts-japanese-mincho | fonts-ipafont-mincho fonts-japanese-gothic
  | fonts-ipafont-gothic fonts-arphic-ukai fonts-arphic-uming fonts-nanum
  docutils-doc fonts-linuxlibertine | ttf-linux-libertine texlive-lang-french
  texlive-latex-base texlive-latex-recommended python-pil-doc python3-pil-dbg
  python-pygments-doc ttf-bitstream-vera sgml-base-doc debhelper
The following NEW packages will be installed:
  awscli docutils-common fontconfig fontconfig-config fonts-dejavu-core
```

```
root@ip-172-31-93-95:~/terraform# aws configure
AWS Access Key ID [None]: AKIAU7PYSBVTHSC6XYNM
AWS Secret Access Key [None]: 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
Default region name [None]:
Default output format [None]: json
root@ip-172-31-93-95:~/terraform#
root@ip-172-31-93-95:~/terraform# terraform plan
```

```
Error: Unsupported argument

on ec2.tf line 4, in resource "aws_instance" "jenkins_instance":
  4:         associate_public_ip_addres = true

An argument named "associate_public_ip_addres" is not expected here. Did you mean
"associate_public_ip_address"?
```

```
root@ip-172-31-93-95:~/terraform# terraform apply
```

```
Error: Unsupported argument

on ec2.tf line 4, in resource "aws_instance" "jenkins_instance":
  4:         associate_public_ip_addres = true

An argument named "associate_public_ip_addres" is not expected here. Did you mean
"associate_public_ip_address"?
```

```
root@ip-172-31-93-95:~/terraform# ls
creds.tf  ec2.tf
root@ip-172-31-93-95:~/terraform# vi ec2.tf
root@ip-172-31-93-95:~/terraform# terraform plan
```

Terraform used the selected providers to generate the following execution plan.
Resource actions are indicated with the following symbols:

```
root@ip-172-31-93-95:~/terraform# ls
creds.tf  ec2.tf
root@ip-172-31-93-95:~/terraform# vi ec2.tf
root@ip-172-31-93-95:~/terraform# terraform plan
```

Terraform used the selected providers to generate the following execution plan.
Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# aws_instance.jenkins_instance will be created
+ resource "aws_instance" "jenkins_instance" {
  + ami                  = "ami-04505e74c0741db8d"
  + arn                  = (known after apply)
  + associate_public_ip_address = true
  + availability_zone     = "us-east-1c"
  + cpu_core_count        = (known after apply)
  + cpu_threads_per_core   = (known after apply)
  + disable_api_termination = (known after apply)
  + ebs_optimized          = (known after apply)
  + get_password_data      = false
  + host_id                = (known after apply)
  + id                    = (known after apply)
  + instance_initiated_shutdown_behavior = (known after apply)
  + instance_state         = (known after apply)
  + instance_type          = "t2.micro"
  + ipv6_address_count      = (known after apply)
  + ipv6_addresses         = (known after apply)
  + key_name                = "hotfix"
  + monitoring              = (known after apply)
  + outpost_arn             = (known after apply)
  + password_data           = (known after apply)
```

```
root@ip-172-31-93-95:~/terraform# terraform apply
```

Terraform used the selected providers to generate the following execution plan.
Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# aws_instance.jenkins_instance will be created
```

```
+ resource "aws_instance" "jenkins_instance" {  
  + ami                  = "ami-04505e74c0741db8d"  
  + arn                  = (known after apply)  
  + associate_public_ip_address = true  
  + availability_zone     = "us-east-1c"  
  + cpu_core_count        = (known after apply)  
  + cpu_threads_per_core   = (known after apply)  
  + disable_api_termination = (known after apply)  
  + ebs_optimized          = (known after apply)  
  + get_password_data      = false  
  + host_id               = (known after apply)  
  + id                    = (known after apply)  
  + instance_initiated_shutdown_behavior = (known after apply)  
  + instance_state         = (known after apply)  
  + instance_type          = "t2.micro"  
  + ipv6_address_count     = (known after apply)  
  + ipv6_addresses         = (known after apply)  
  + key_name               = "hotfix"  
  + monitoring             = (known after apply)  
  + outpost_arn            = (known after apply)  
  + password_data          = (known after apply)  
  + placement_group        = (known after apply)  
  + placement_partition_number = (known after apply)  
  + primary_network_interface_id = (known after apply)  
}
```



```
root@ip-172-31-93-95:~/terraform# vi ec2.tf
root@ip-172-31-93-95:~/terraform# terraform apply
aws_instance.jenkins_instance: Refreshing state... [id=i-0c61552dd80389b2d]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

-/+ destroy and then create replacement

Terraform will perform the following actions:

```
# aws_instance.jenkins_instance must be replaced
-/+ resource "aws_instance" "jenkins_instance" {
  ~ ami                        = "ami-04505e74c0741db8d" -> "ami-0e472ba40eb589f49" # forces replacement
  ~ arn                      = "arn:aws:ec2:us-east-1:342508047718:instance/i-0c61552dd80389b2d" -> (known
after apply)
  ~ cpu_core_count           = 1 -> (known after apply)
  ~ cpu_threads_per_core     = 1 -> (known after apply)
  ~ disable_api_termination  = false -> (known after apply)
  ~ ebs_optimized            = false -> (known after apply)
  - hibernation              = false -> null
  + host_id                  = (known after apply)
  ~ id                      = "i-0c61552dd80389b2d" -> (known after apply)
  ~ instance_initiated_shutdown_behavior = "stop" -> (known after apply)
  ~ instance_state           = "running" -> (known after apply)
  ~ ipv6_address_count       = 0 -> (known after apply)
  ~ ipv6_addresses          = [] -> (known after apply)
  ~ monitoring               = false -> (known after apply)
  + outpost_arn              = (known after apply)
  + password_data            = (known after apply)
  + placement_group          = (known after apply)
  + placement_partition_number = (known after apply)
  ~ primary_network_interface_id = "eni-08d600550cf9ca831" -> (known after apply)
  ~ private_dns              = "ip-172-31-86-127.ec2.internal" -> (known after apply)
```


