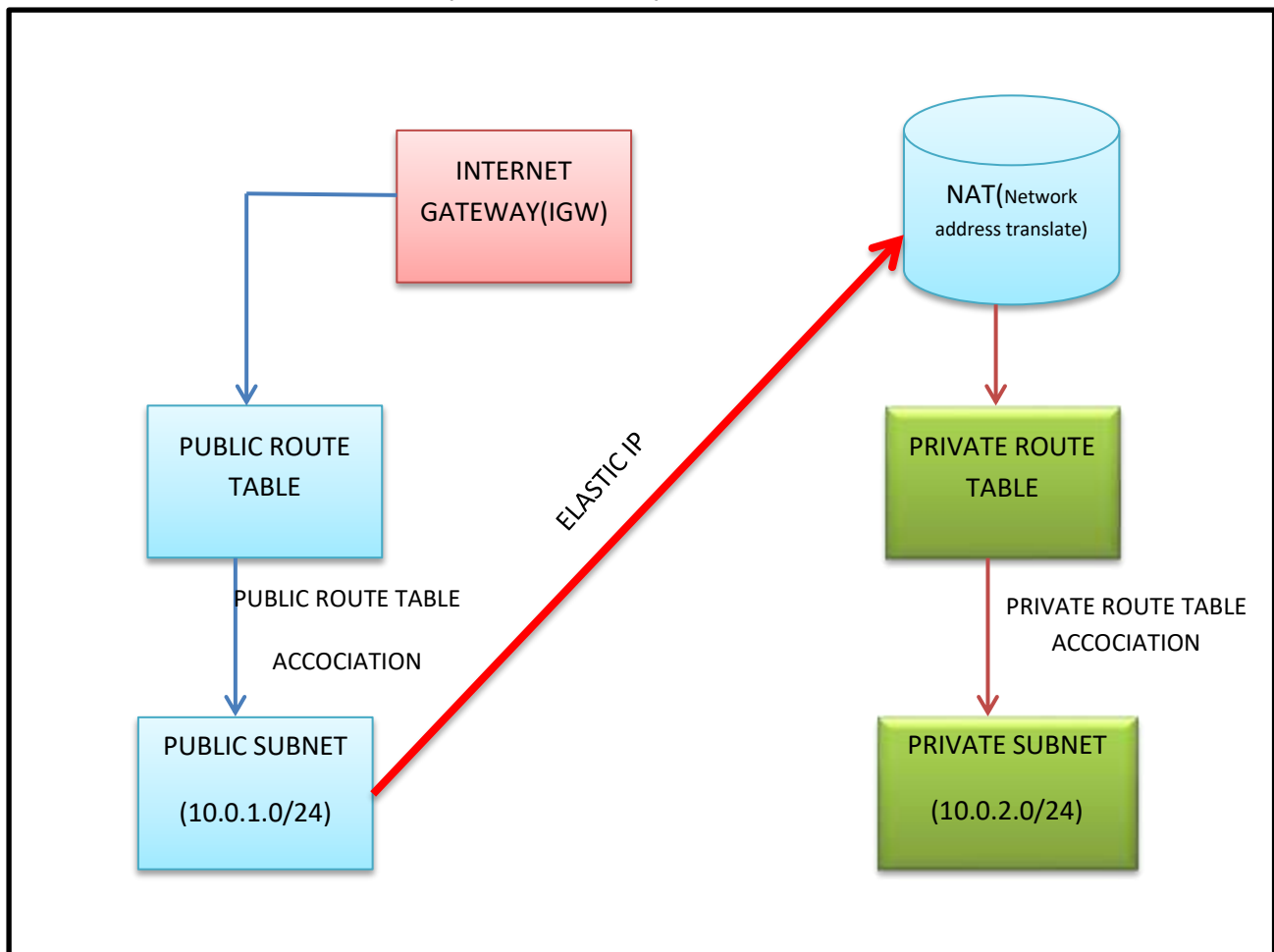


# **TERRAFORM**

Terraform is a tool for building, changing and versioning infrastructure safely and efficiently.

Provider=aws

MY VPC(10.0.0.0/16) =>65535



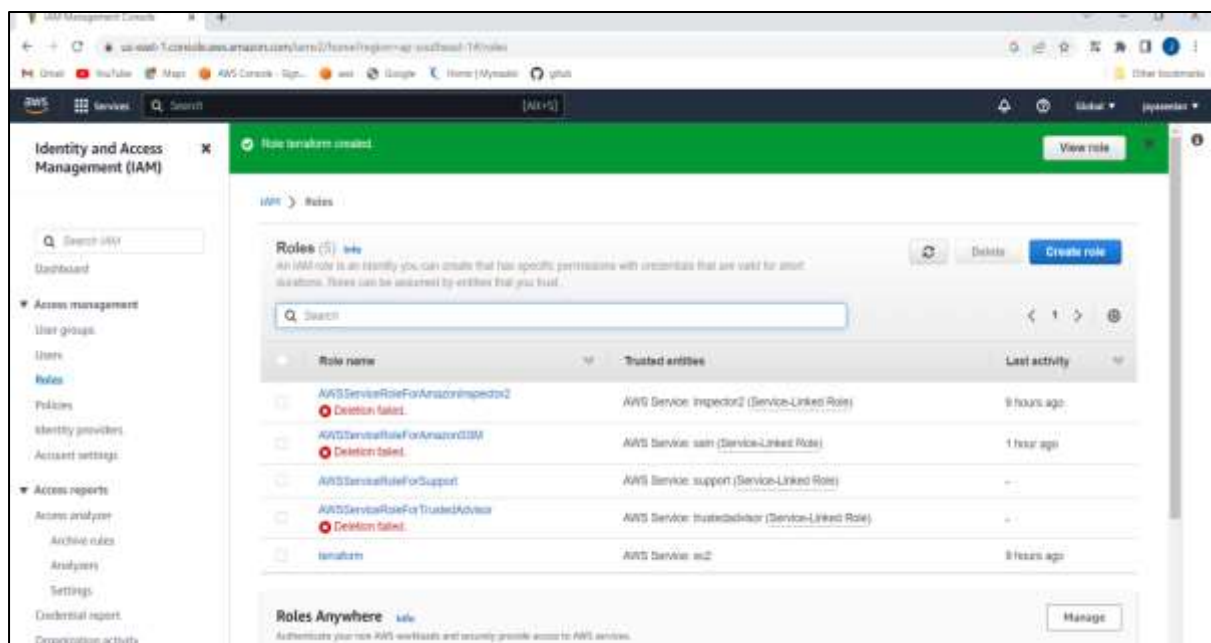
## **Steps Terraform**

- 1.Provider
- 1.2. VPC
- 2.Public subnet
- 3.Private subnet
- 4.IGW
- 5.Public Route table
- 6.Public Routetable association
- 7.EIP
- 8.NAT
- 9.Private Route table
- 10.Private Route table association
- 11.Public sg
- 12.Private sg
- 13.Public instance
- 14.Private instance

## **Steps To Create Terraform**

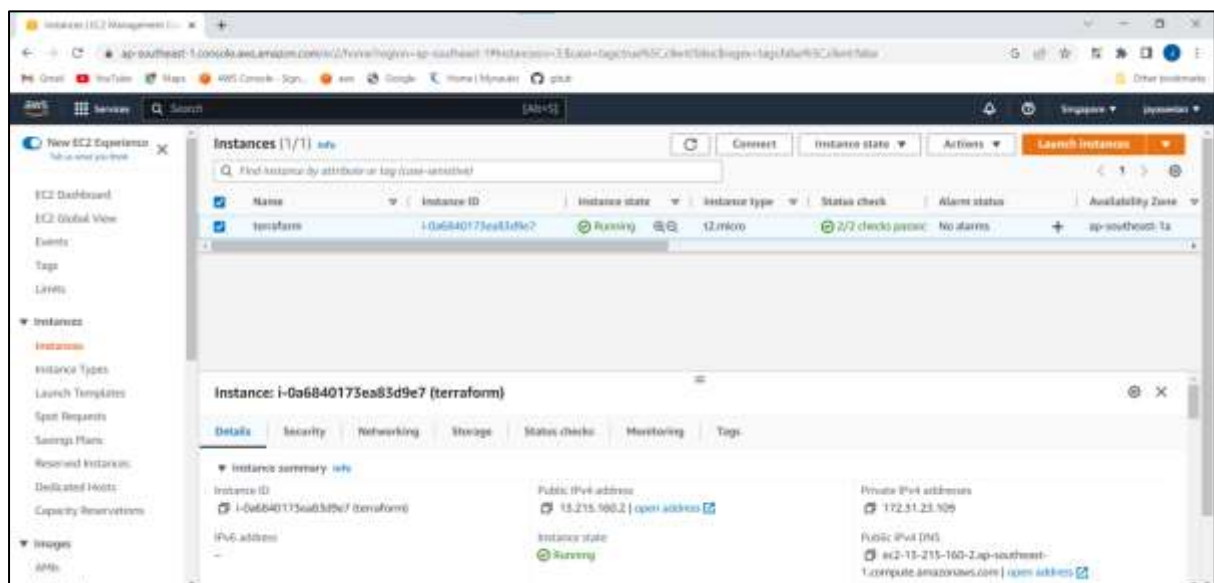
### **Step 1:Create Iam Role**

IAM --->Create Role --->aws service --->ec2 service --->next --->policy(AdministratorAcces)--->next--->rolename(terraform)--->create role.



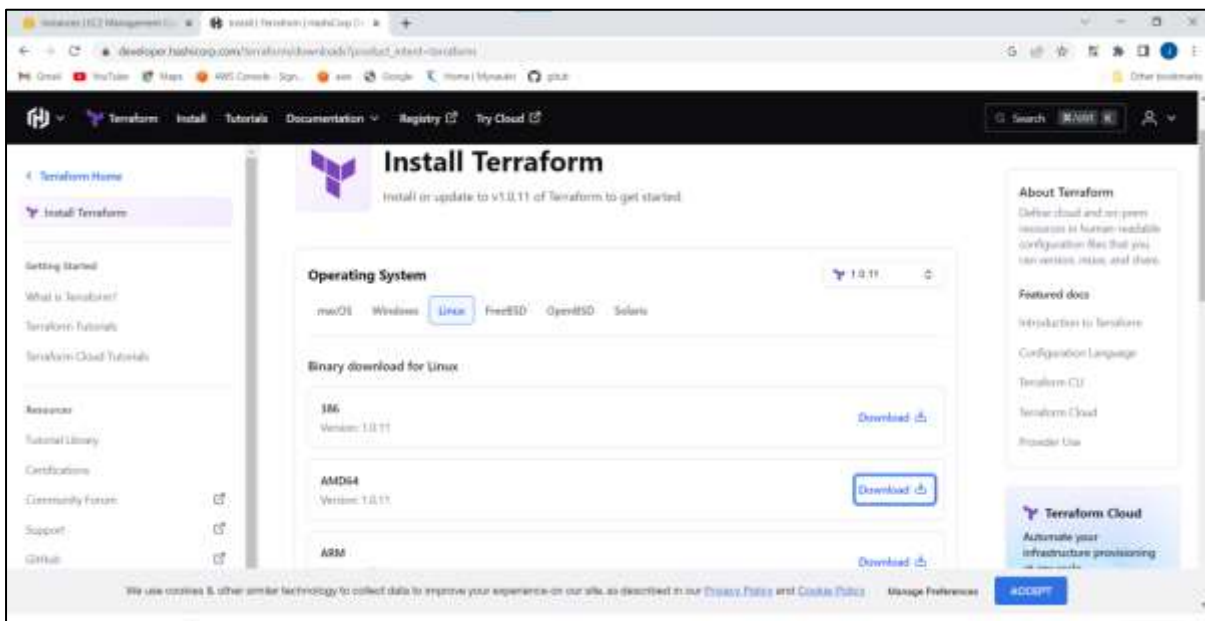
## Step2:Create Ec2 Ubuntu Server

Version(18.04)---->key(.pem)----->security(all tcp)---  
 >storage(30gb)--->advance(role attach) --->launch instance



## Step3:Terraform Download

Google --->terraform.io---> Terraform by HashiCorp--->click right corner download--->operating system(linux)--->version(1.0.11)--  
 >Amd 64--->download (right click copy)



Now server login and download terraform

Pem key convert ppk then--->Login server(Ubuntu) --->#sudo -i --  
--->

#wget terraform link

#wget https://releases.hashicorp.com/terraform/1.0.11/terraform\_1.0.11\_linux\_amd64.zip

```

see "man sudo_root" for details.

ubuntu@ip-172-31-23-109:~$ sudo -i
root@ip-172-31-23-109:~# wget https://releases.hashicorp.com/terraform/1.0.11/terraform_1.0.11_linux_amd64.zip
--2023-02-14 05:16:45-- https://releases.hashicorp.com/terraform/1.0.11/terraform_1.0.11_linux_amd64.zip
Resolving releases.hashicorp.com (releases.hashicorp.com)... 18.135.68.53, 18.15
1.68.68, 18.155.68.87, ...
Connecting to releases.hashicorp.com (releases.hashicorp.com)|18.155.68.53|:443.
.. connected.
HTTP request sent, awaiting response... 200 OK
Length: 10082446 (10M) [application/zip]
Saving to: 'terraform_1.0.11_linux_amd64.zip'

terraform_1.0.11_linux_amd64.zip [100%]=====> 17.24M 63.69M/s in 0.1s

2023-02-14 05:16:45 (63.6 MB/s) = 'terraform_1.0.11_linux_amd64.zip' saved (1008
2446/10082446)

root@ip-172-31-23-109:~# ll
total 17608
-rwxr-xr-x 1 root root 4096 Feb 14 05:16 ./
-rwxr-xr-x 1 root root 4096 Feb 14 05:04 ../
-rw-r--r-- 1 root root 3104 Apr 9 2019 .bashrc
-rw-r--r-- 1 root root 148 Aug 17 2015 .gnupg
-rwxr-xr-x 1 root root 4096 Feb 14 05:04 .ssh/
-rw-r--r-- 1 root root 177 Feb 14 05:16 .wget-hsts
-rwxr-xr-x 1 root root 10082446 May 9 2022 terraform_1.0.11_linux_amd64.zip
root@ip-172-31-23-109:~#

```

Now install unzip --->(because unzip is not default Ubuntu)

#apt-get install unzip

#unzip terraform\_1.0.11\_linux\_amd64.zip

# rm -rv terraform\_1.0.11\_linux\_amd64.zip

#ll --->shown unzip terraform

```

Processing triggers for mine-support (3.60ubuntu1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
root@ip-172-31-23-109:~# unzip terraform_1.0.11_linux_amd64.zip
Archive:  terraform_1.0.11_linux_amd64.zip
  inflating: terraform
root@ip-172-31-23-109:~# ll
total 75388
drwx----- 4 root root    4096 Feb 14 05:19 ./
drwxr-xr-x 23 root root    4096 Feb 14 05:04 ../
-rw-r--r-- 1 root root    3106 Apr  9 2018 .bashrc
-rw-r--r-- 1 root root     148 Aug 17 2015 .profile
drwx----- 2 root root    4096 Feb 14 05:04 .ssh/
-rw-r--r-- 1 root root     177 Feb 14 05:16 .wget-hsts
drwx----- 3 root root    4096 Feb 14 05:05 snap/
-rwxr-xr-x 1 root root 59084800 Nov 10 2021 terraform*
-rw-r--r-- 1 root root 18082446 May  9 2022 terraform_1.0.11_linux_amd64.zip
root@ip-172-31-23-109:~# rm -rv terraform_1.0.11_linux_amd64.zip
removed 'terraform_1.0.11_linux_amd64.zip'
root@ip-172-31-23-109:~# ll
total 57728
drwx----- 4 root root    4096 Feb 14 05:19 ./
drwxr-xr-x 23 root root    4096 Feb 14 05:04 ../
-rw-r--r-- 1 root root    3106 Apr  9 2018 .bashrc
-rw-r--r-- 1 root root     148 Aug 17 2015 .profile
drwx----- 2 root root    4096 Feb 14 05:04 .ssh/
-rw-r--r-- 1 root root     177 Feb 14 05:16 .wget-hsts
drwx----- 3 root root    4096 Feb 14 05:05 snap/
-rwxr-xr-x 1 root root 59084800 Nov 10 2021 terraform*
root@ip-172-31-23-109:~#

```

Now terraform move /usr/local/bin/

```
#mv terraform /usr/local/bin/
```

```
#cd /usr/local/bin/
```

```
#ll --->shown terraform
```

Terraform version check

```
#terraform -version --->it will shown version
```

```

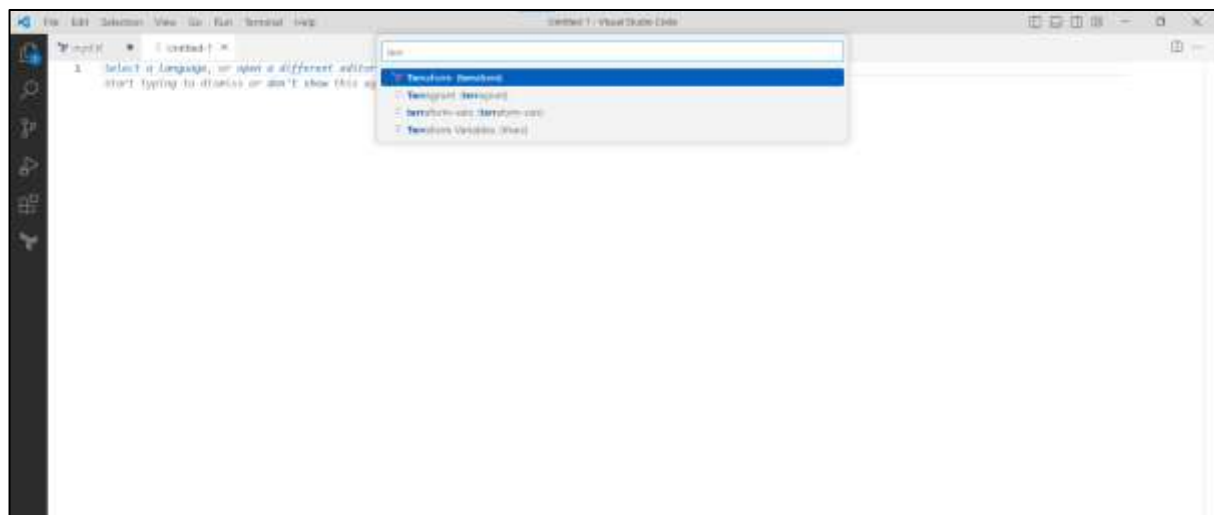
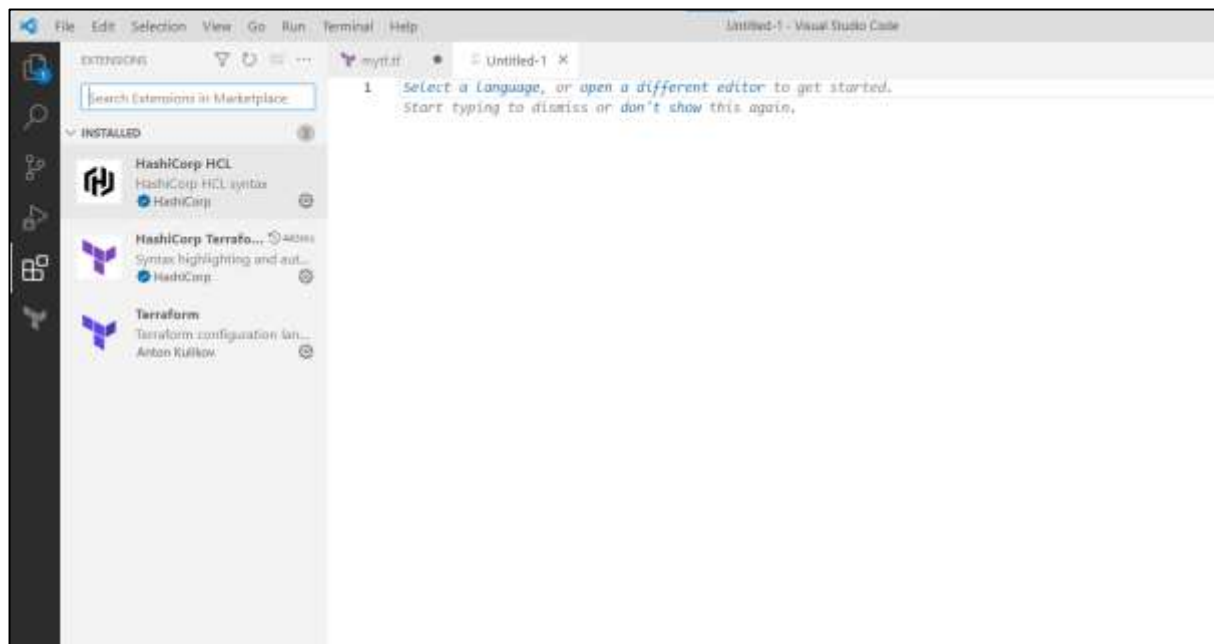
drwx----- 3 root root    4096 Feb 14 05:05 snap/
-rwxr-xr-x 1 root root 59084800 Nov 10 2021 terraform*
-rw-r--r-- 1 root root 18082446 May  9 2022 terraform_1.0.11_linux_amd64.zip
root@ip-172-31-23-109:~# rm -rv terraform_1.0.11_linux_amd64.zip
removed 'terraform_1.0.11_linux_amd64.zip'
root@ip-172-31-23-109:~# ll
total 57728
drwx----- 4 root root    4096 Feb 14 05:19 ./
drwxr-xr-x 23 root root    4096 Feb 14 05:04 ../
-rw-r--r-- 1 root root    3106 Apr  9 2018 .bashrc
-rw-r--r-- 1 root root     148 Aug 17 2015 .profile
drwx----- 2 root root    4096 Feb 14 05:04 .ssh/
-rw-r--r-- 1 root root     177 Feb 14 05:16 .wget-hsts
drwx----- 3 root root    4096 Feb 14 05:05 snap/
-rwxr-xr-x 1 root root 59084800 Nov 10 2021 terraform*
root@ip-172-31-23-109:~# mv terraform /usr/local/bin/
root@ip-172-31-23-109:~# cd /usr/local/bin/
root@ip-172-31-23-109:~# ll
total 57704
drwxr-xr-x 2 root root    4096 Feb 14 05:23 ./
drwxr-xr-x 18 root root    4096 Jan 11 15:49 ../
-rwxr-xr-x 1 root root 59084800 Nov 10 2021 terraform*
root@ip-172-31-23-109:~# terraform --version
Terraform v1.0.11
on linux_amd64

Your version of Terraform is out of date! The latest version
is 1.3.8. You can update by downloading from https://www.terraform.io/downloads.html
root@ip-172-31-23-109:~#

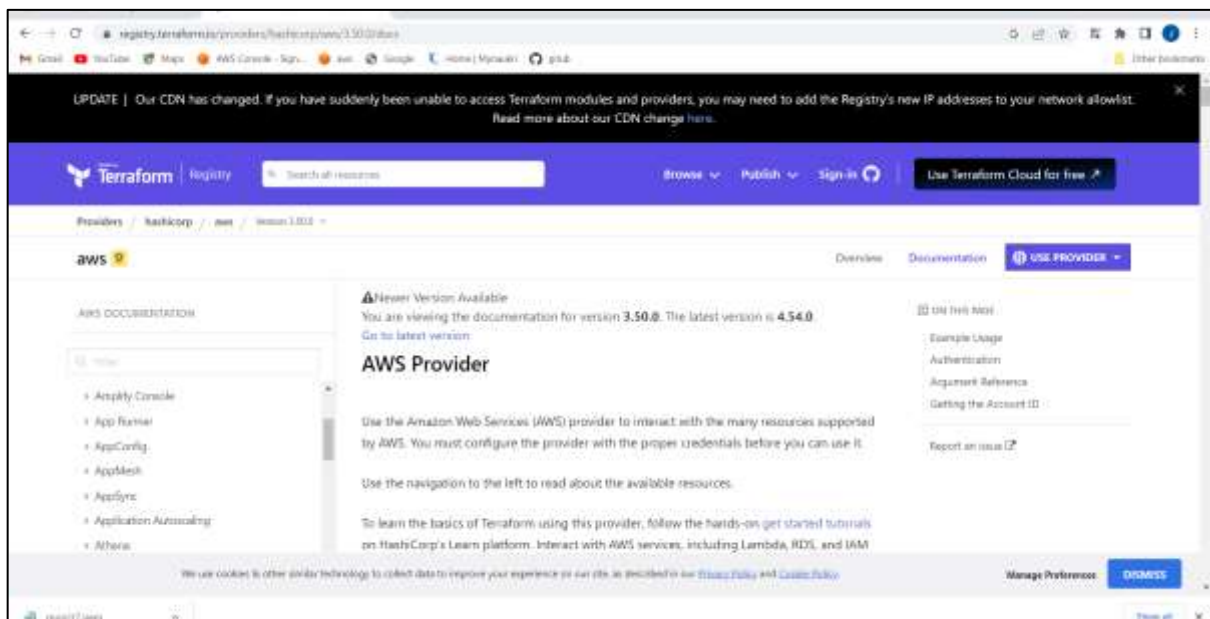
```

## Step4:Create Code

Download visual studio code --->file--->new file --->select language(terraform) -->suppose (can't show terraform, download harshicorp hcl and harsicorp terraform) --->save (ctrl +s)--->then write script

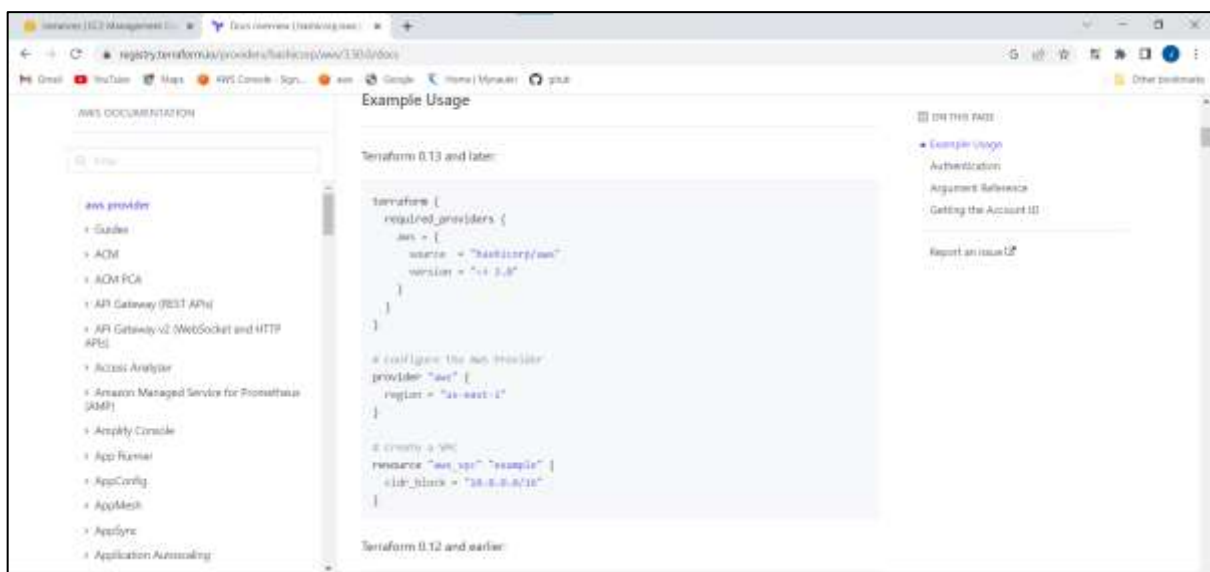


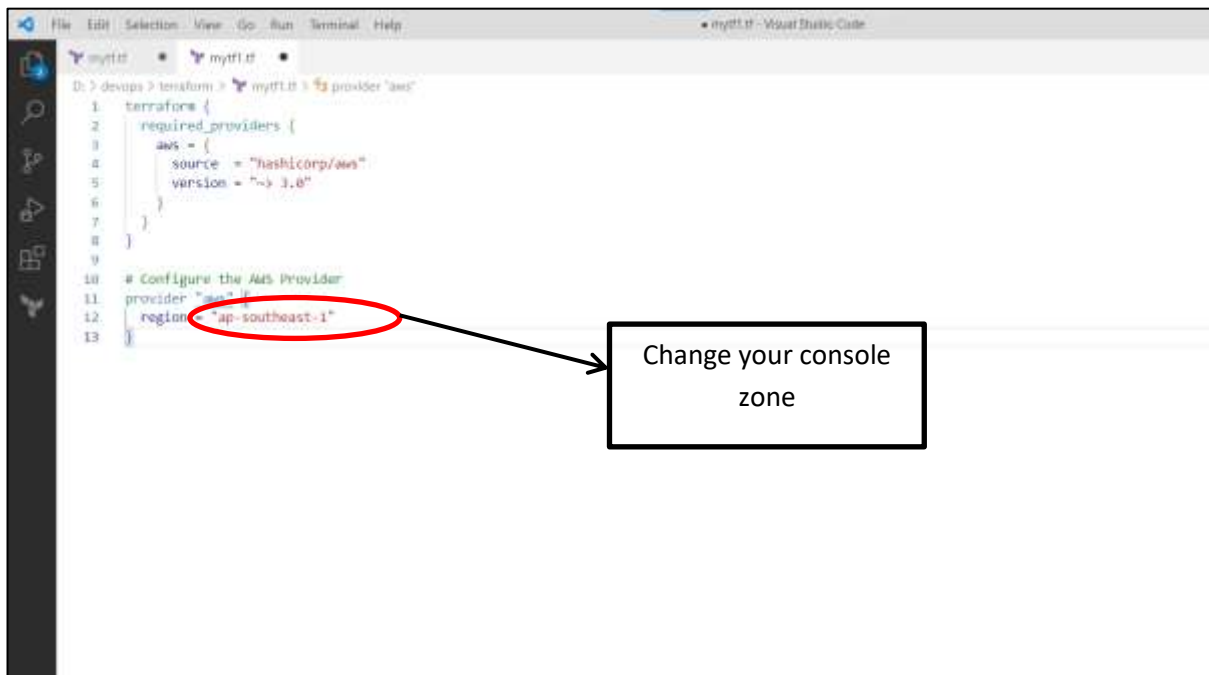
Which place to take code --->google --->terraform.io--->aws provider--->version select(3.50.0)



## Step5:Write Code For Provider

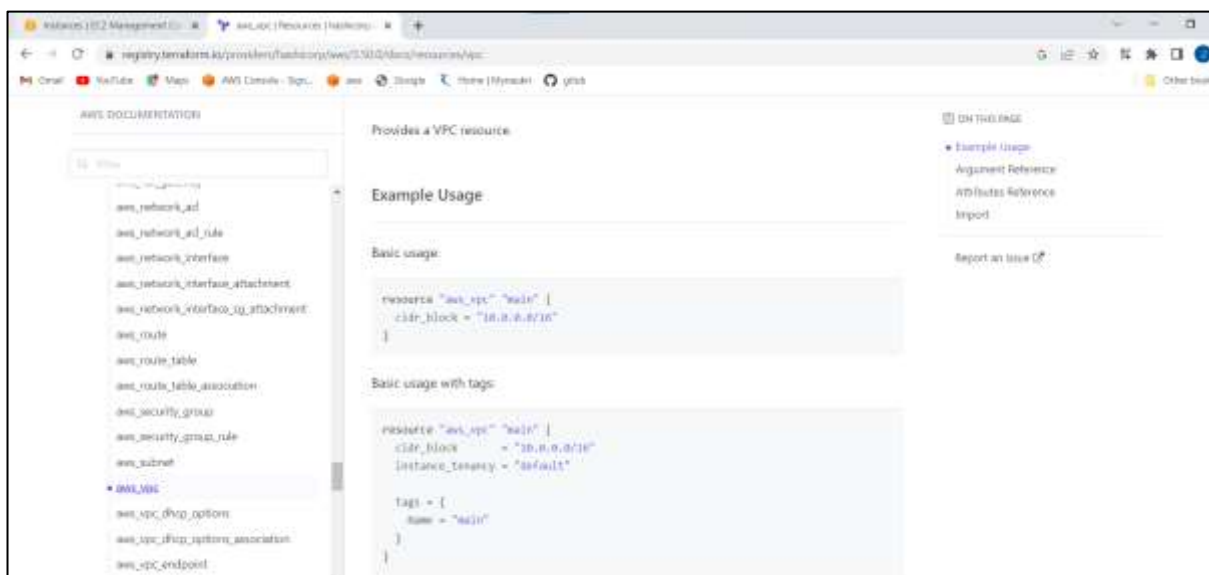
Left side select aws provider --->copy code and put visual studio code -->



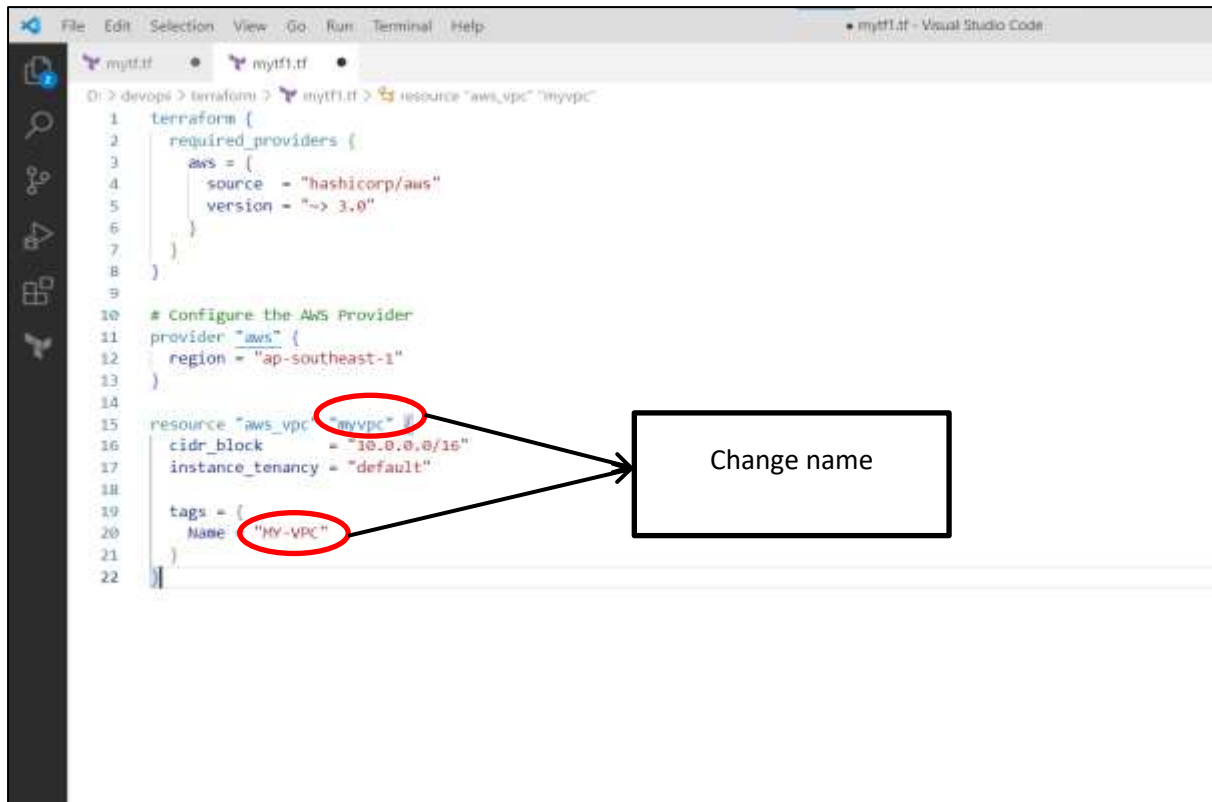


## Step6: Write Code For vpc

Left side select vpc--->select aws vpc --->copy code and put visual studio code -->

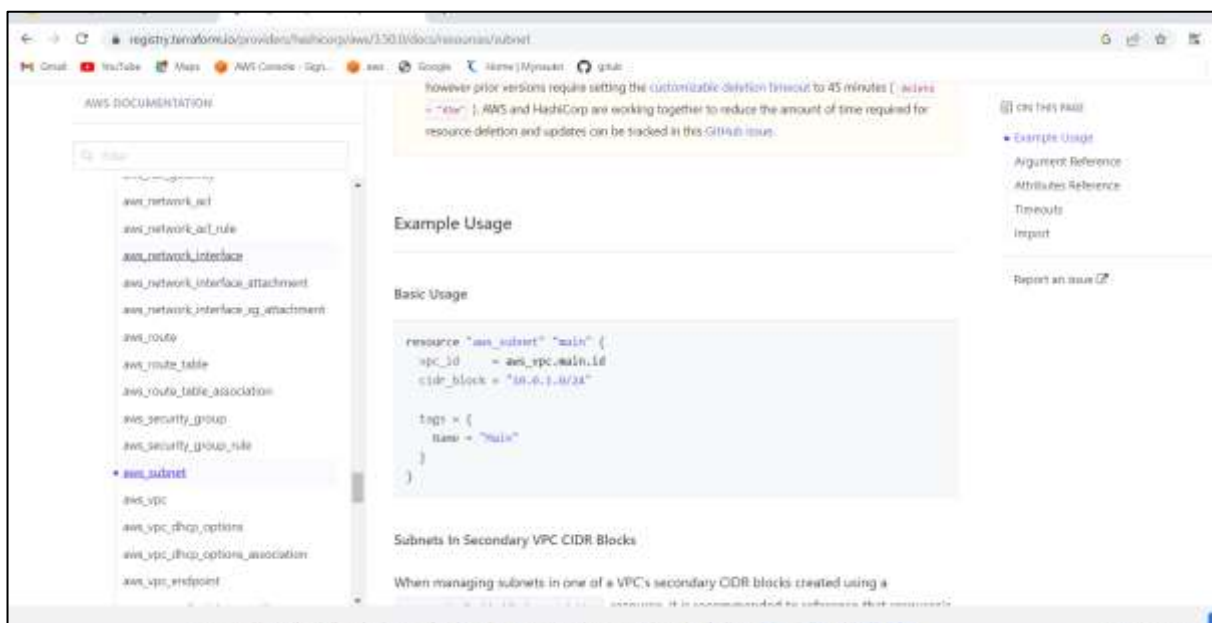


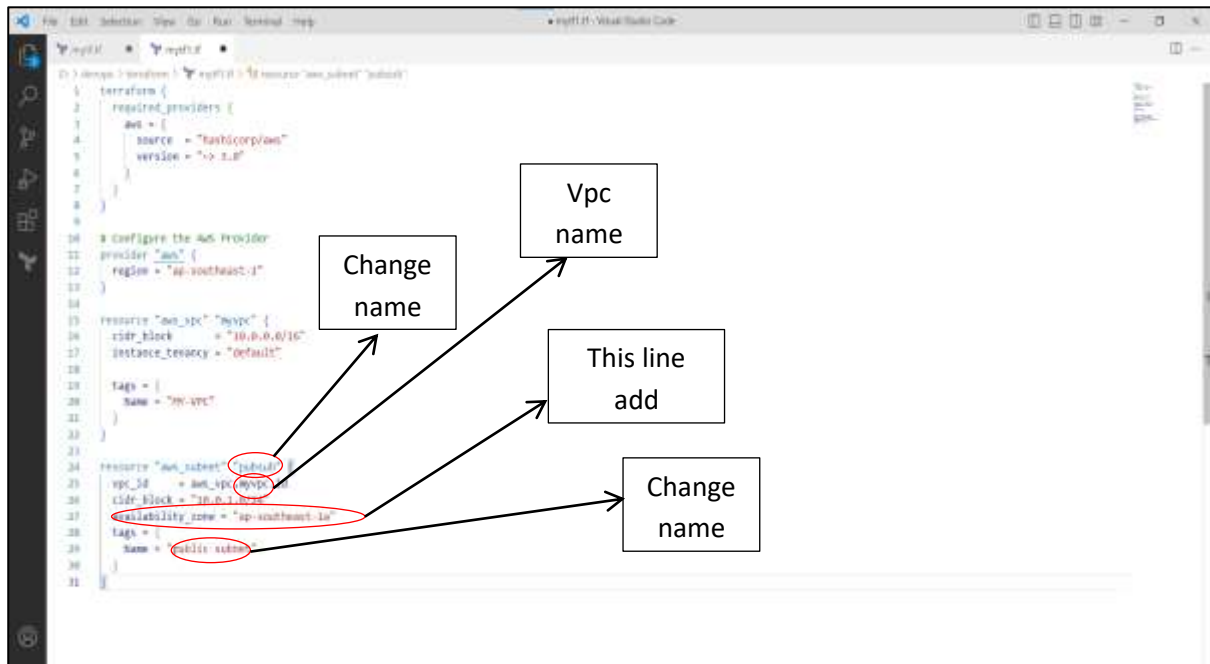




## Step7: Write Code For public Subnet

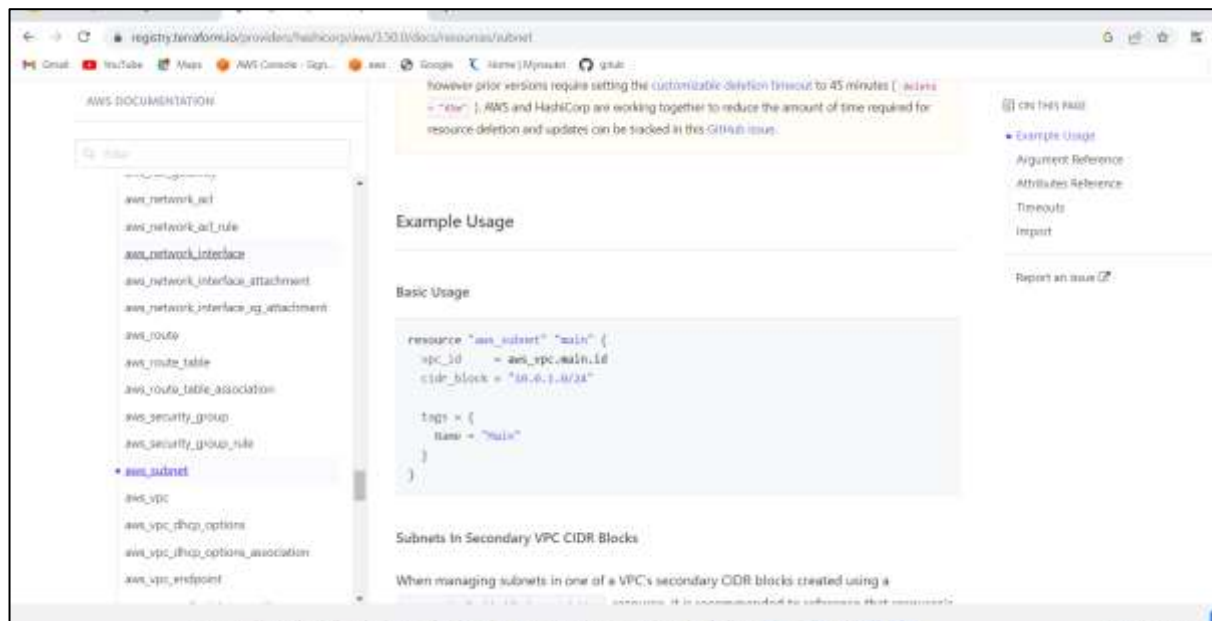
Left side select vpc--->aws subnet --->copy code and put visual studio code -->

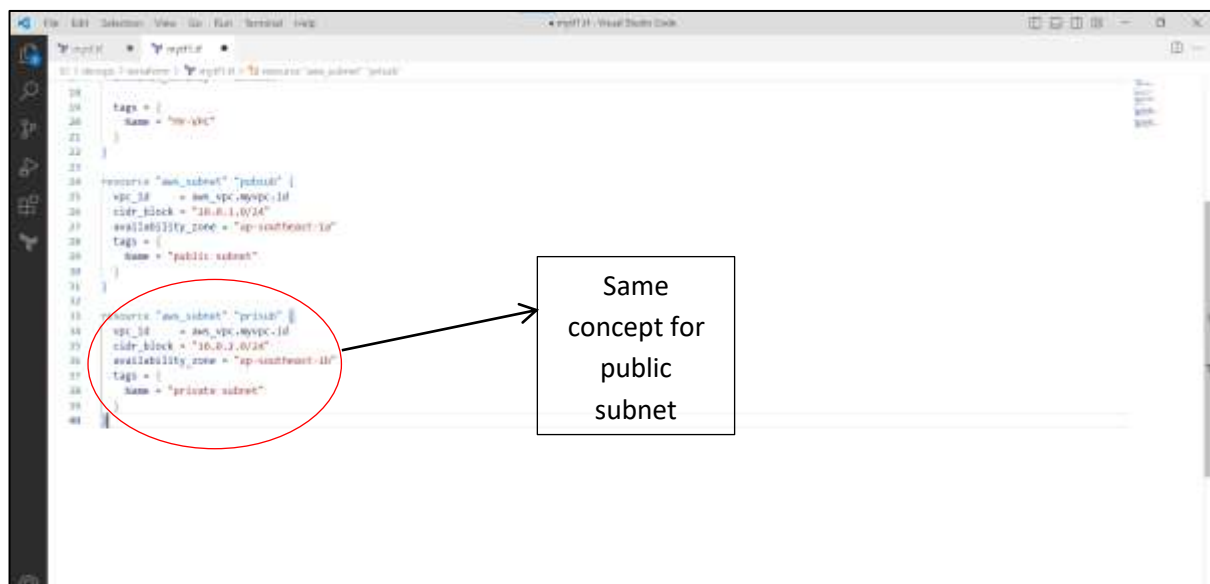




## Step8: Write Code For private Subnet

Left side select vpc--->aws subnet --->copy code and put visual studio code -->

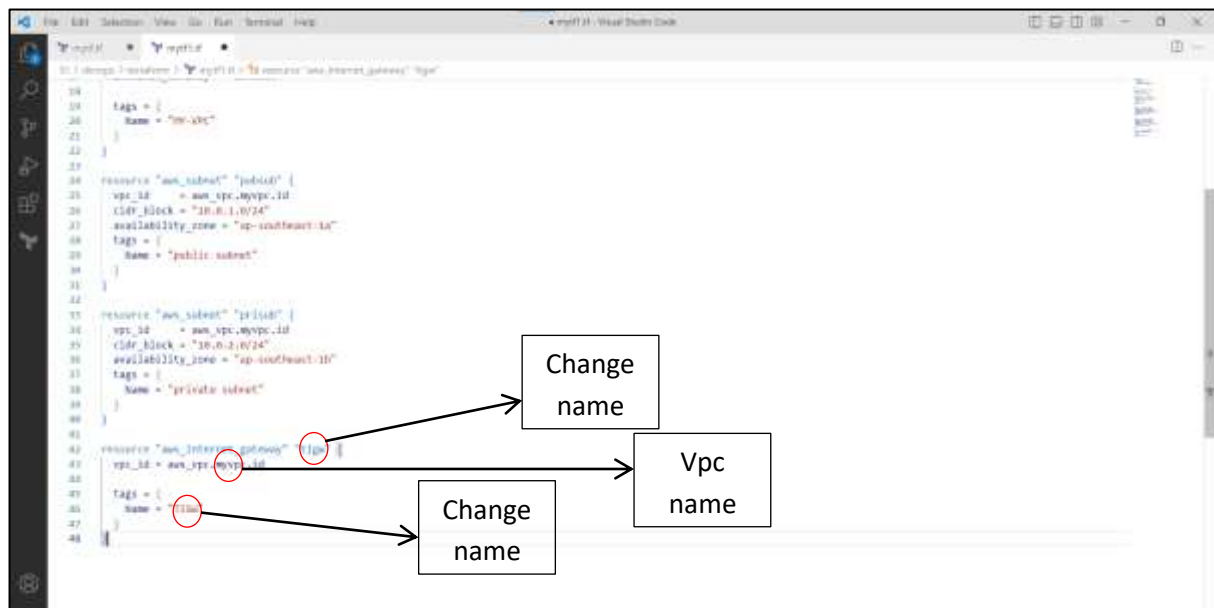




## Step9: Write Code For Internet Gateway

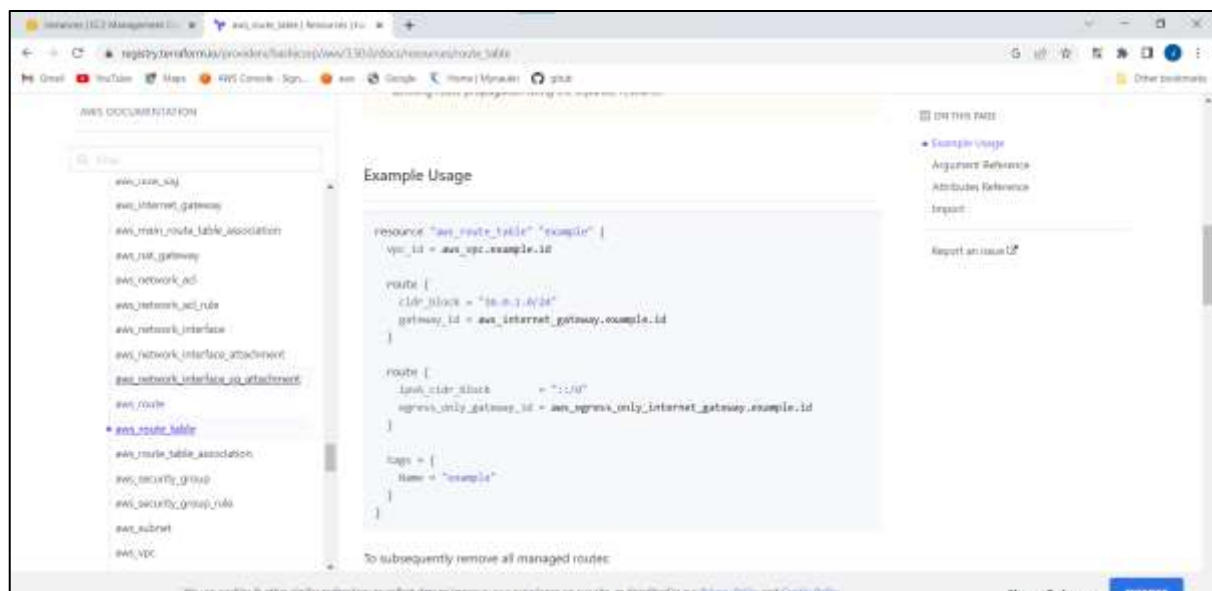
Left side select vpc--->aws internet gateway --->copy code and put visual studio code -->

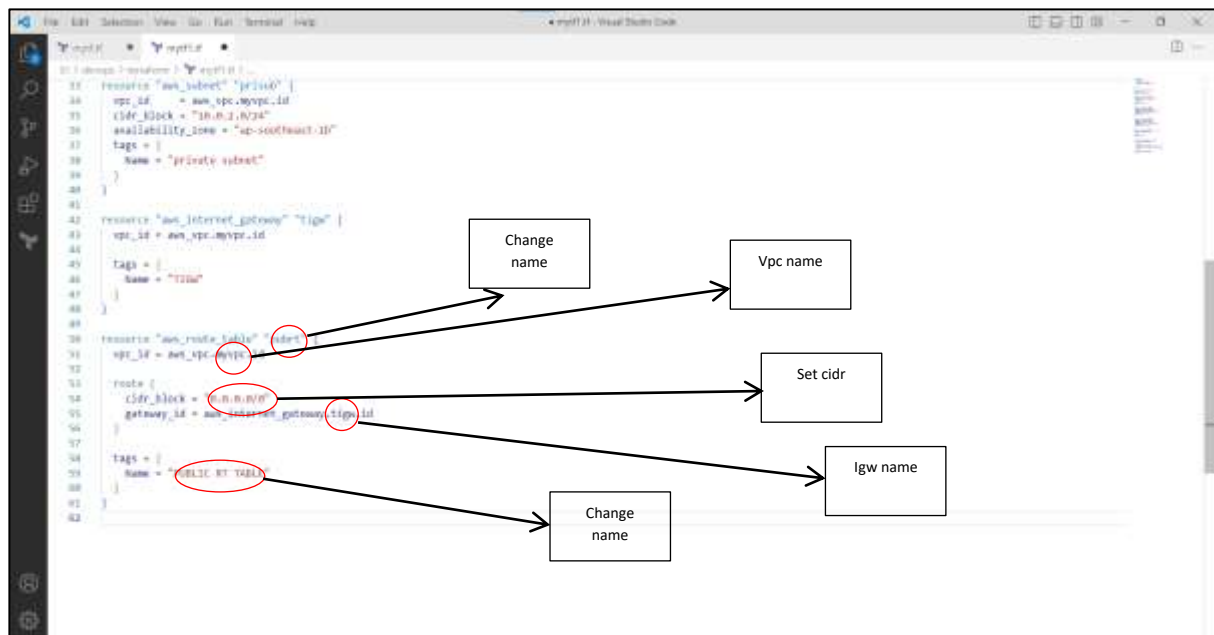




## Step10: Write Code For Public Route Table

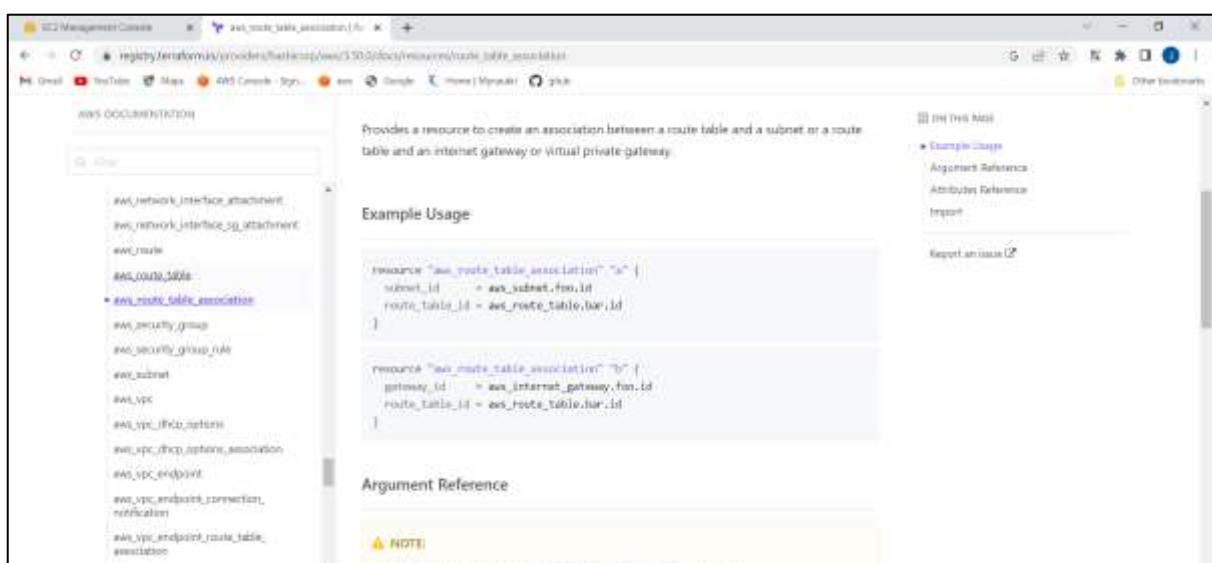
Left side select vpc--->aws route table --->copy code and put visual studio code -->

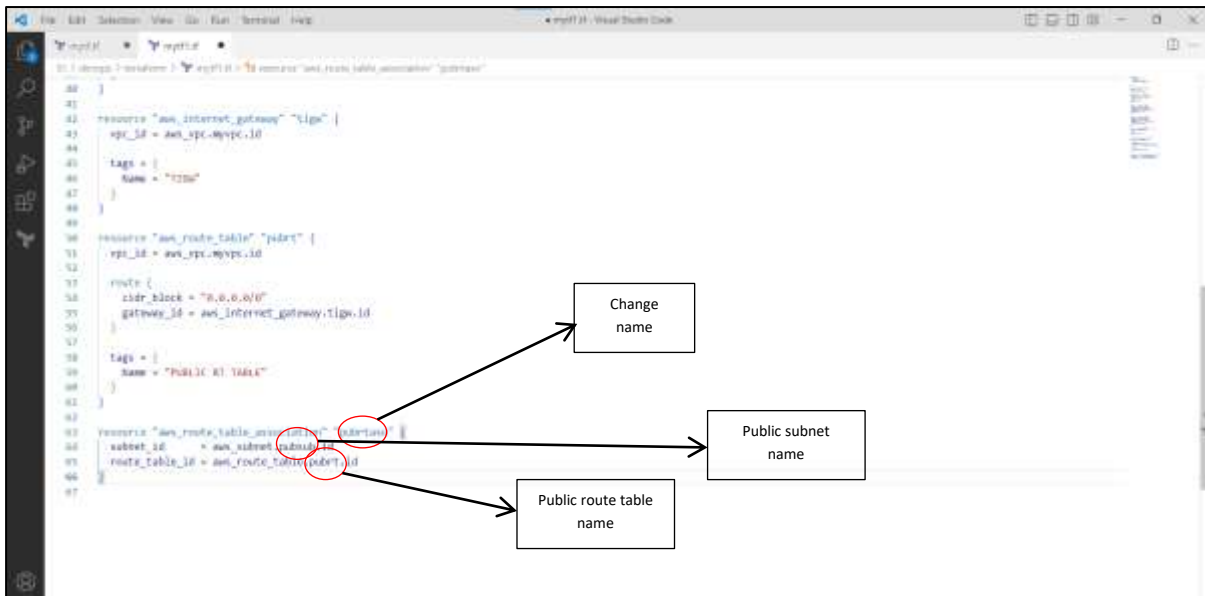




## Step11: Write Code For Public Route Table Association

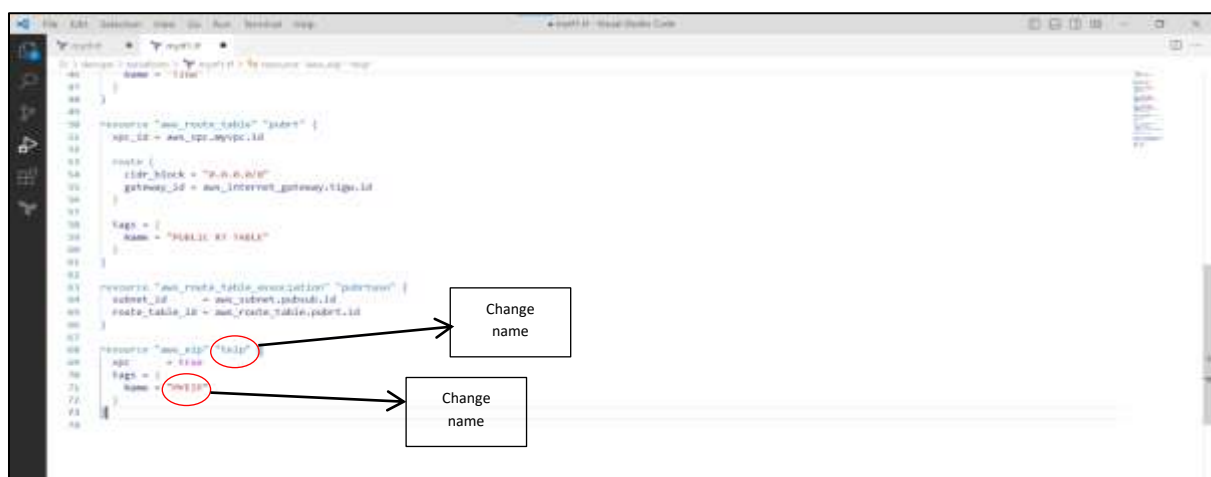
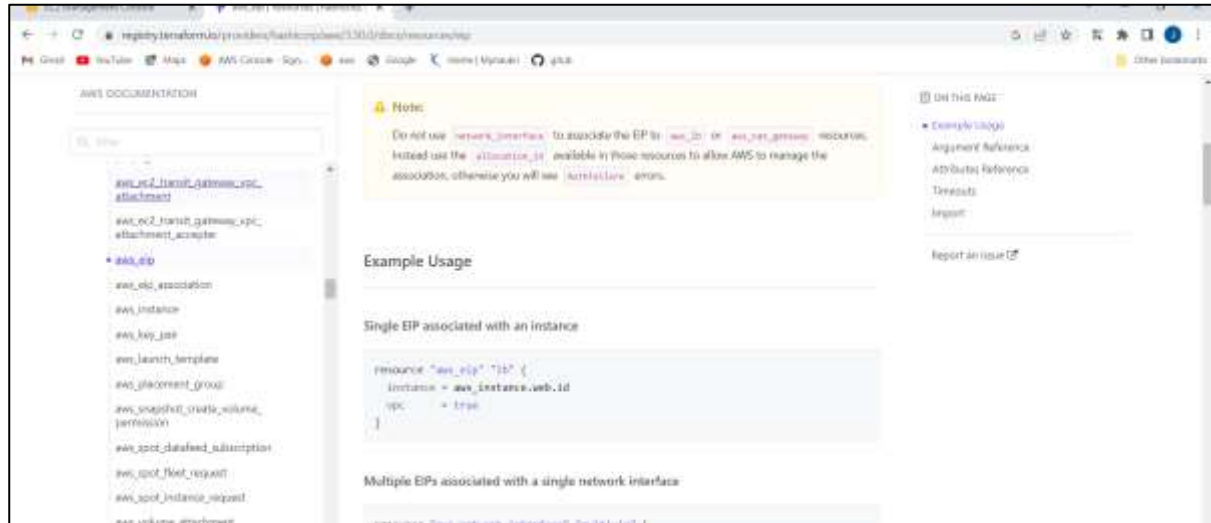
Left side select vpc--->aws route table Association --->copy code and put visual studio code -->





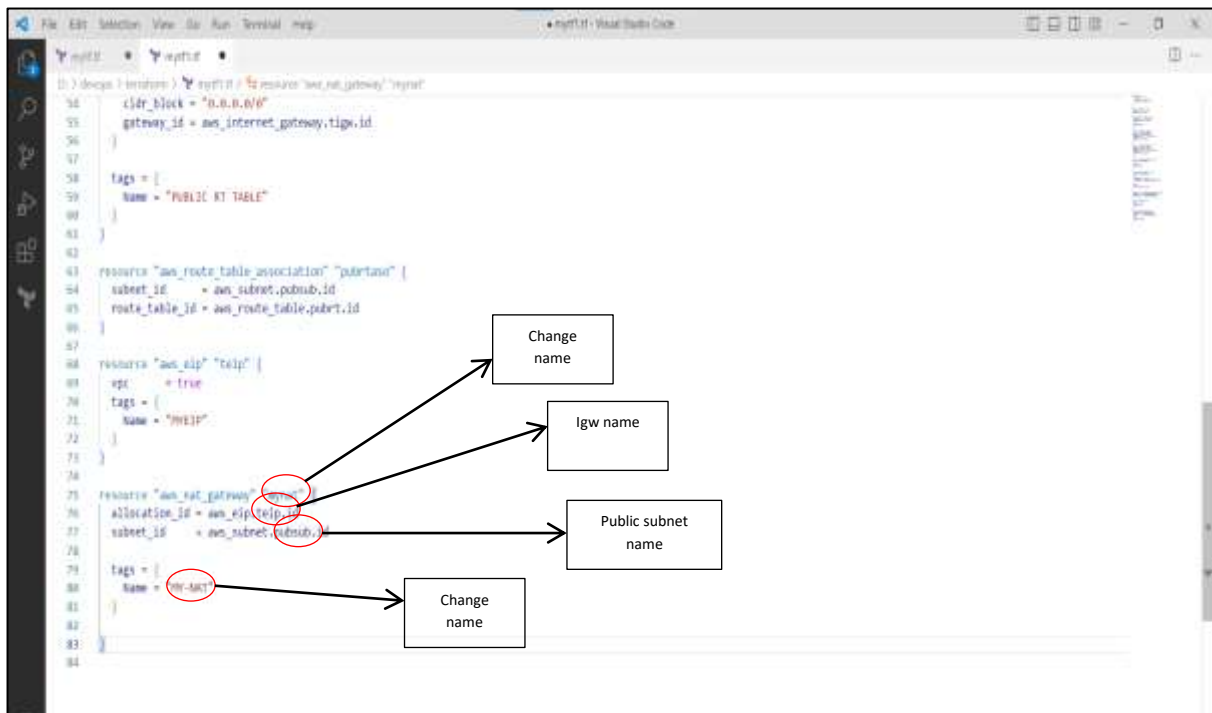
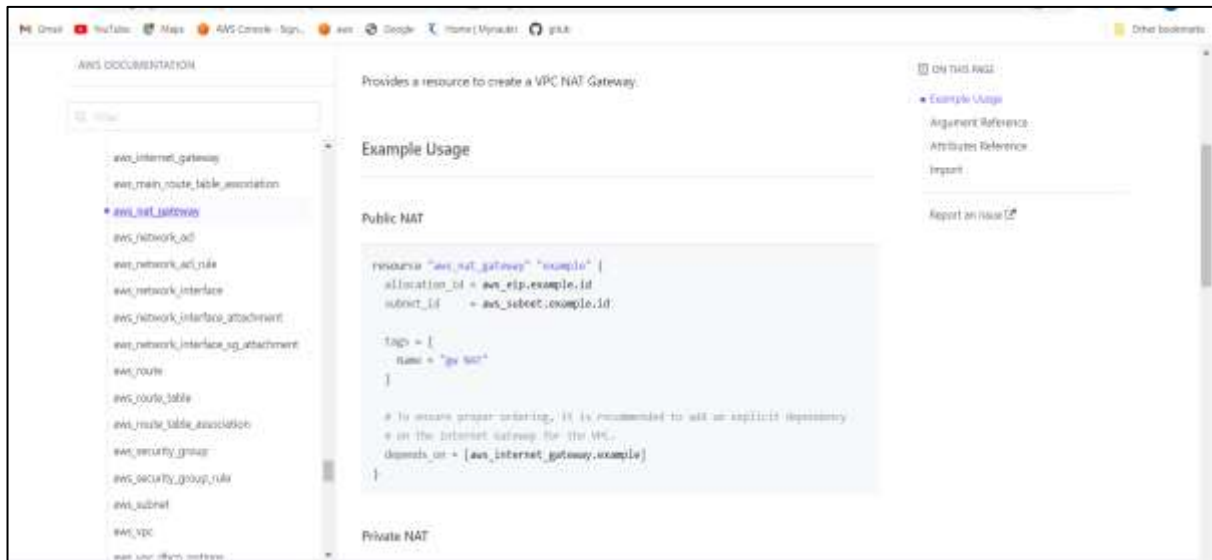
## Step12: Write Code For Elastic Ip

Left side select ec2--->aws elastic ip --->copy code and put visual studio code -->



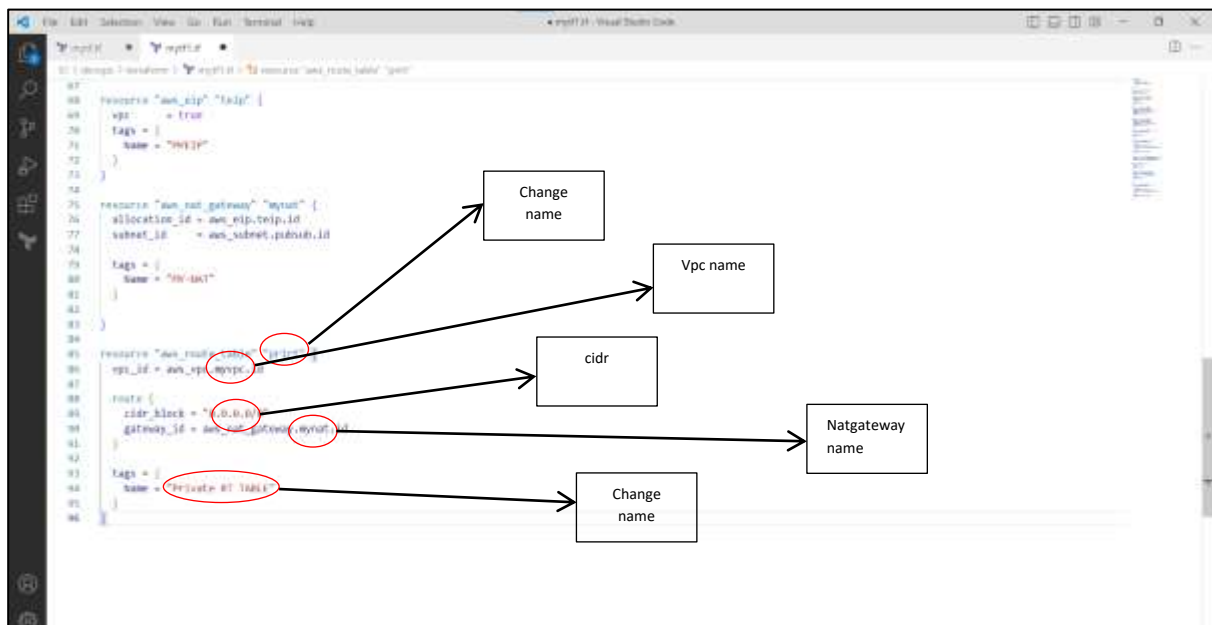
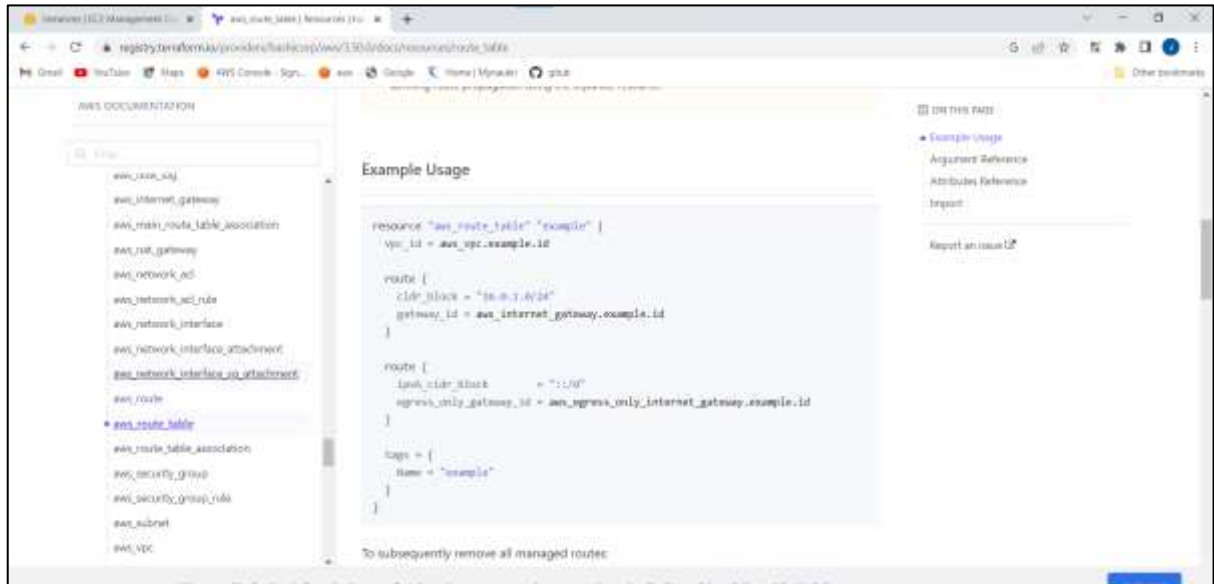
## Step13: Write Code For Nat Gateway

Left side select vpc--->aws nat gateway --->copy code and put visual studio code -->



## Step14: Write Code For Private Route Table

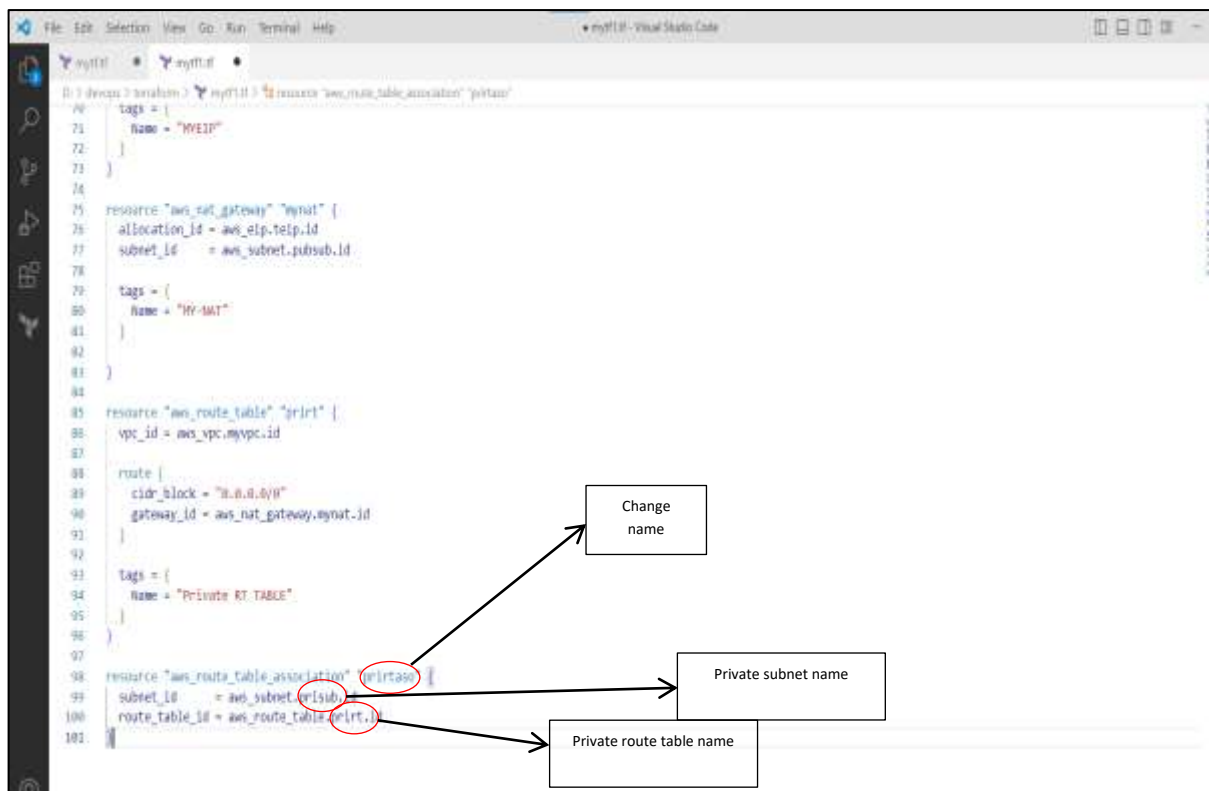
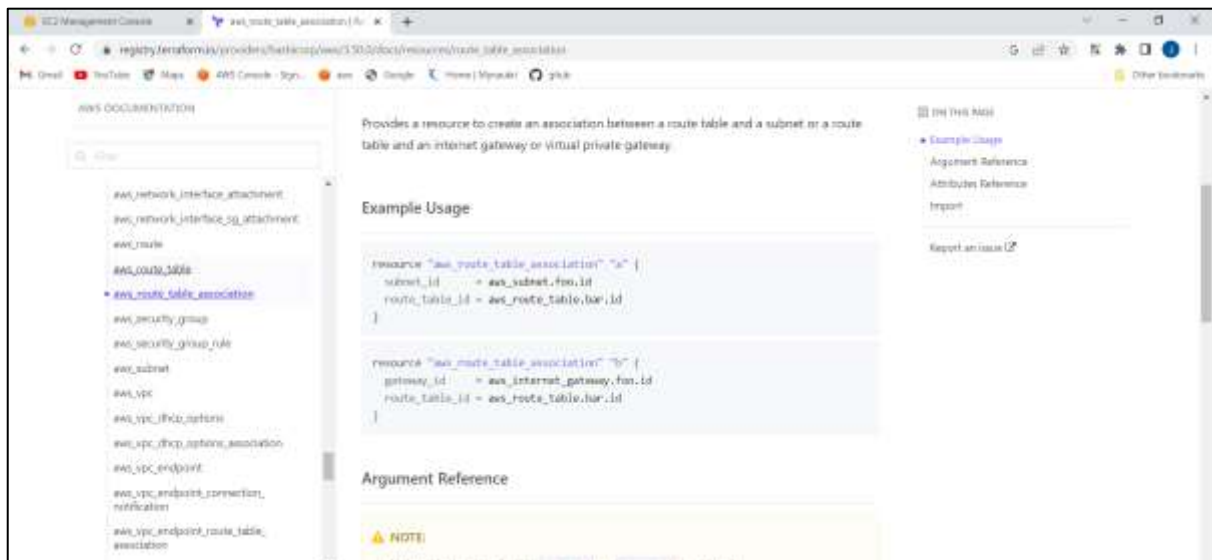
Left side select vpc--->aws route table --->copy code and put visual studio code -->





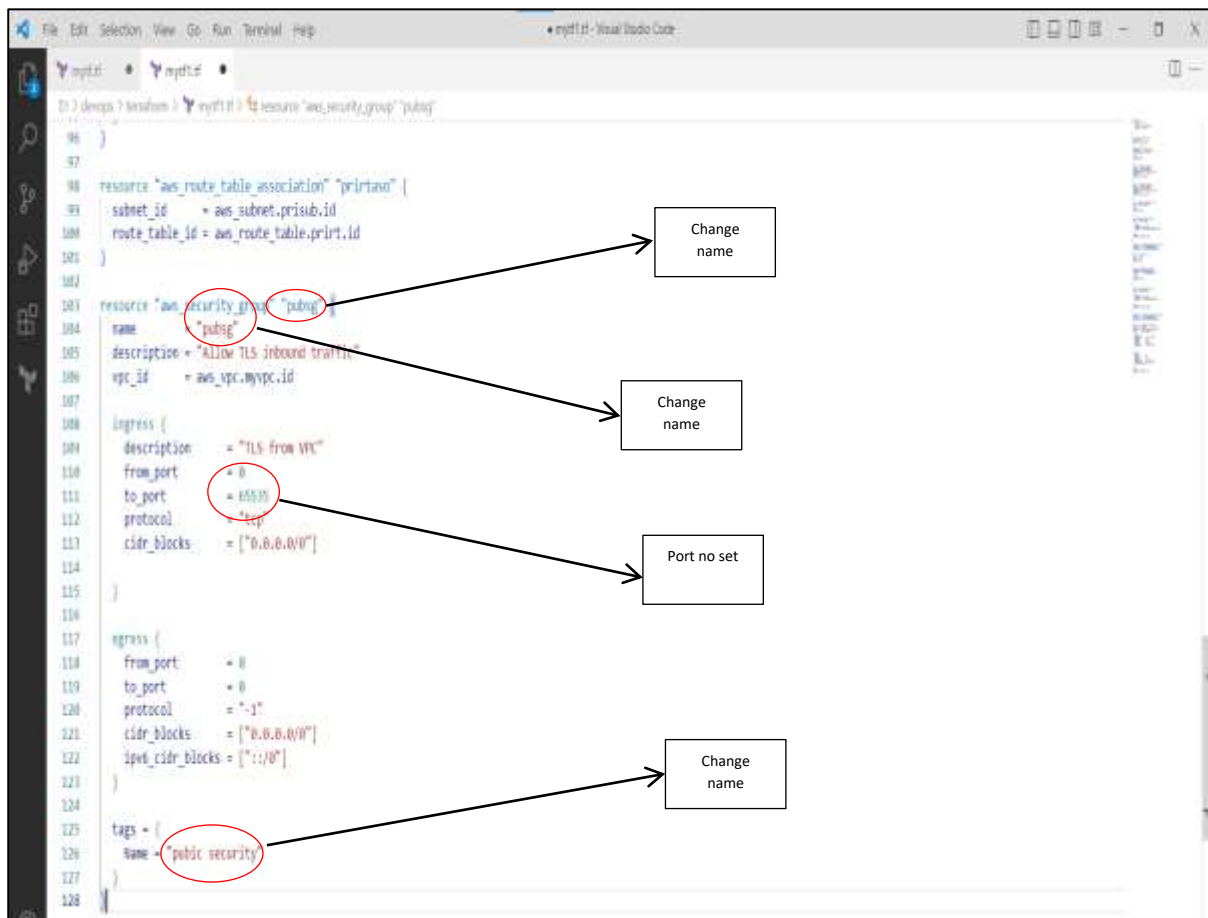
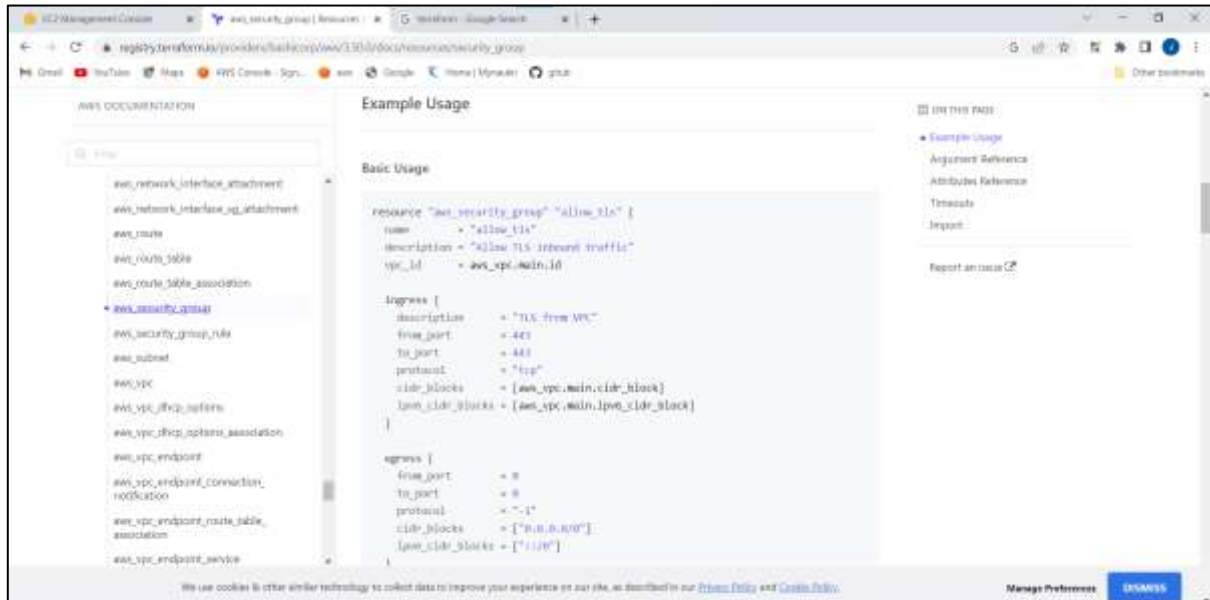
## Step15: Write Code For Private Route Table Association

Left side select vpc--->aws route table Association --->copy code and put visual studio code -->



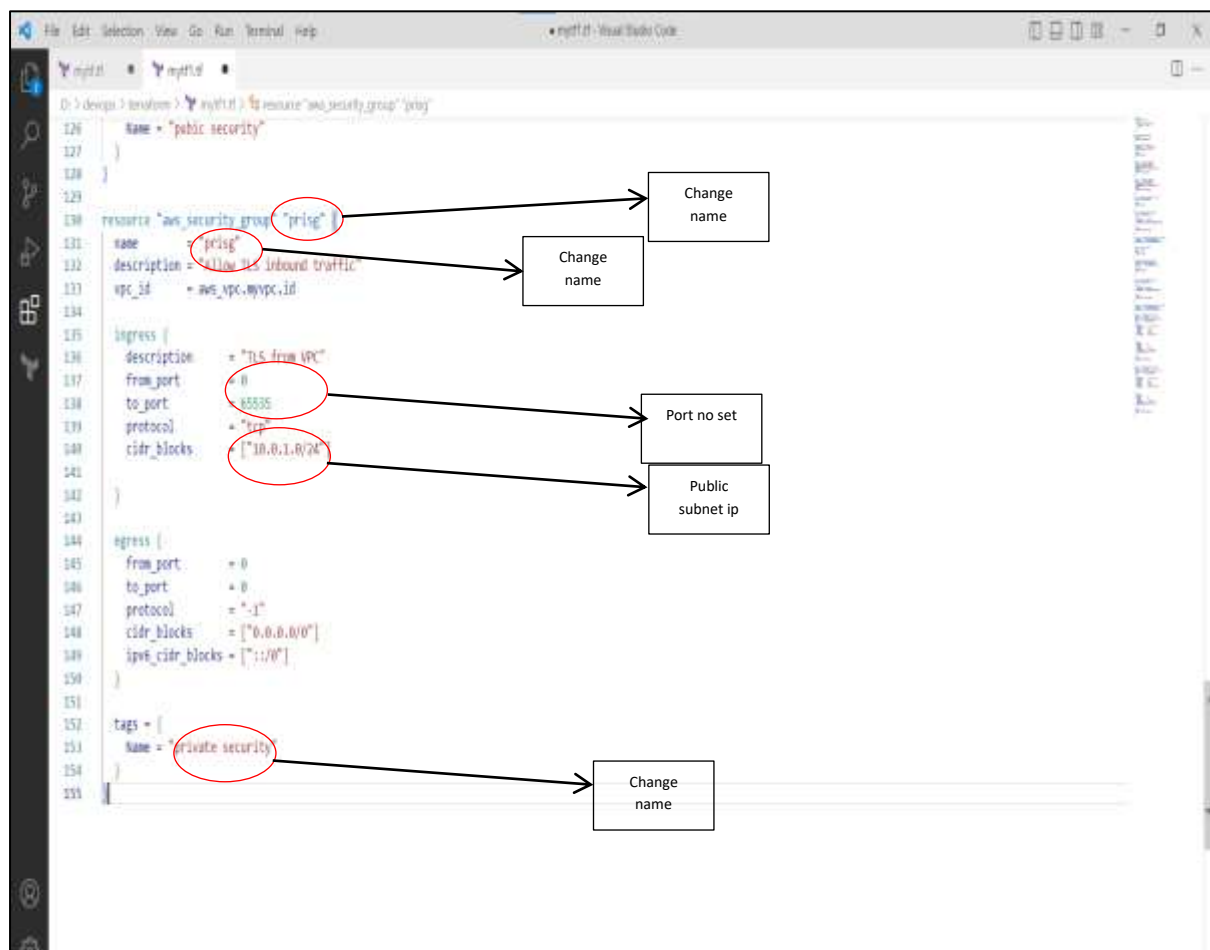
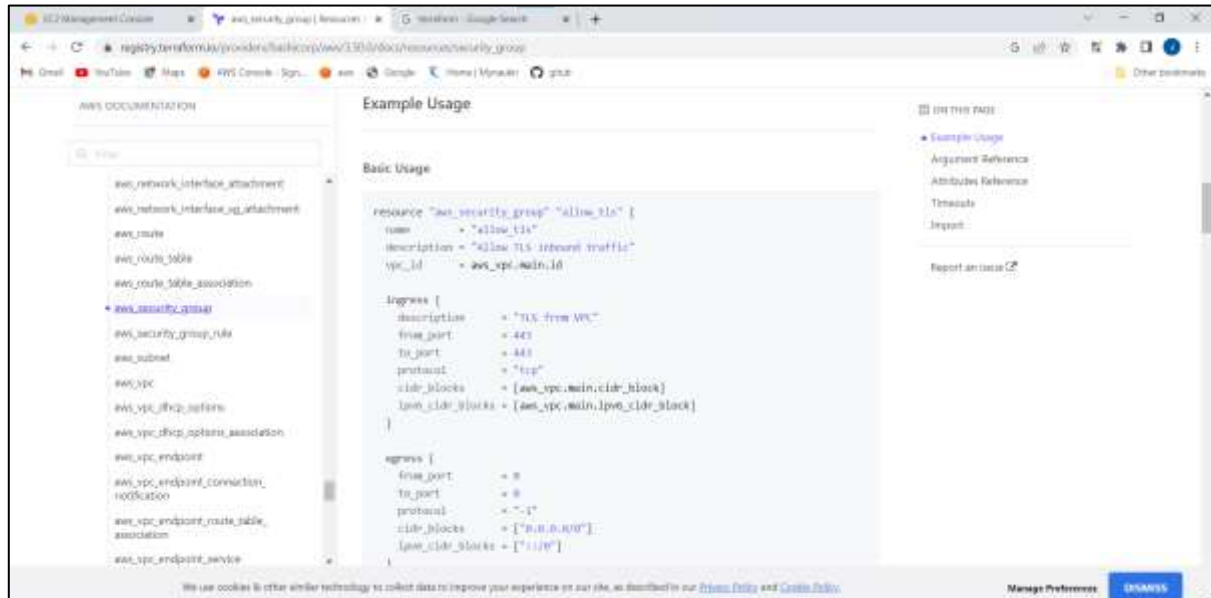
## Step16: Write Code For public security

Left side select vpc---> aws security--->copy code and put visual studio code -->



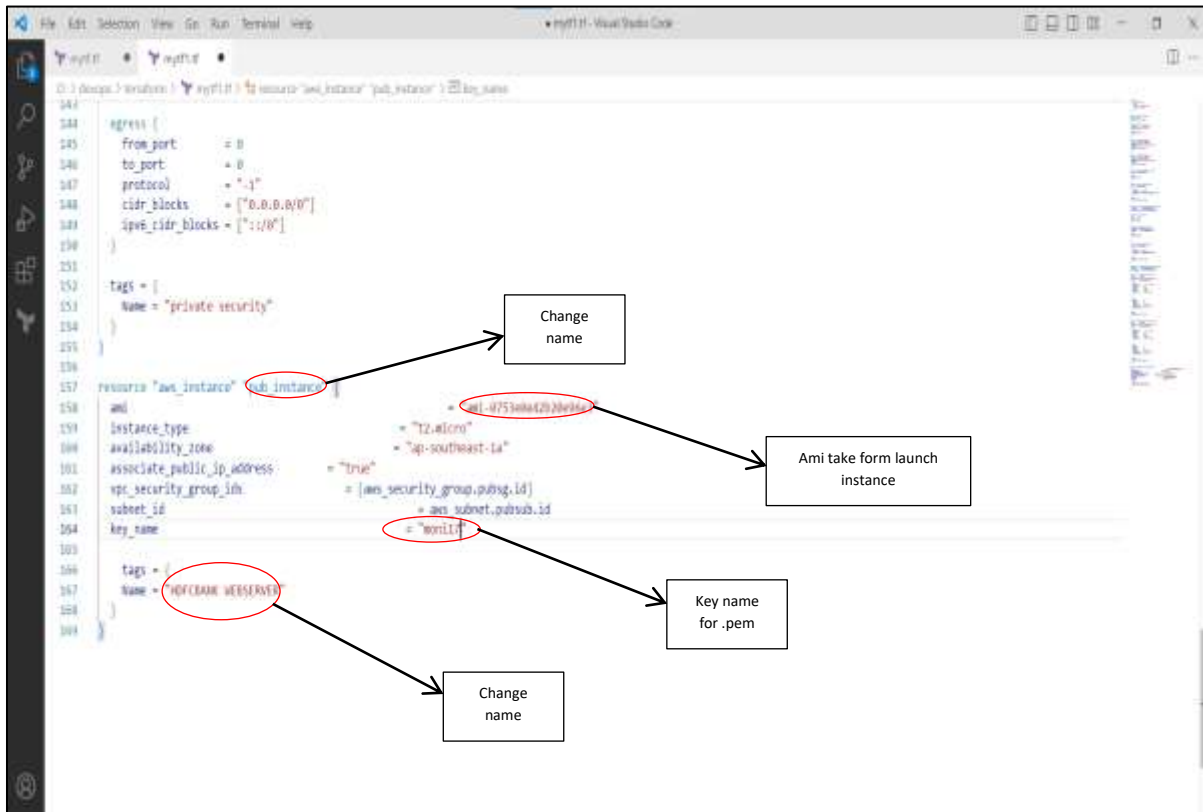
## Step17: Write Code For private security

Left side select vpc---> aws security--->copy code and put visual studio code -->



## Step18: Write Code For public ec2 instance

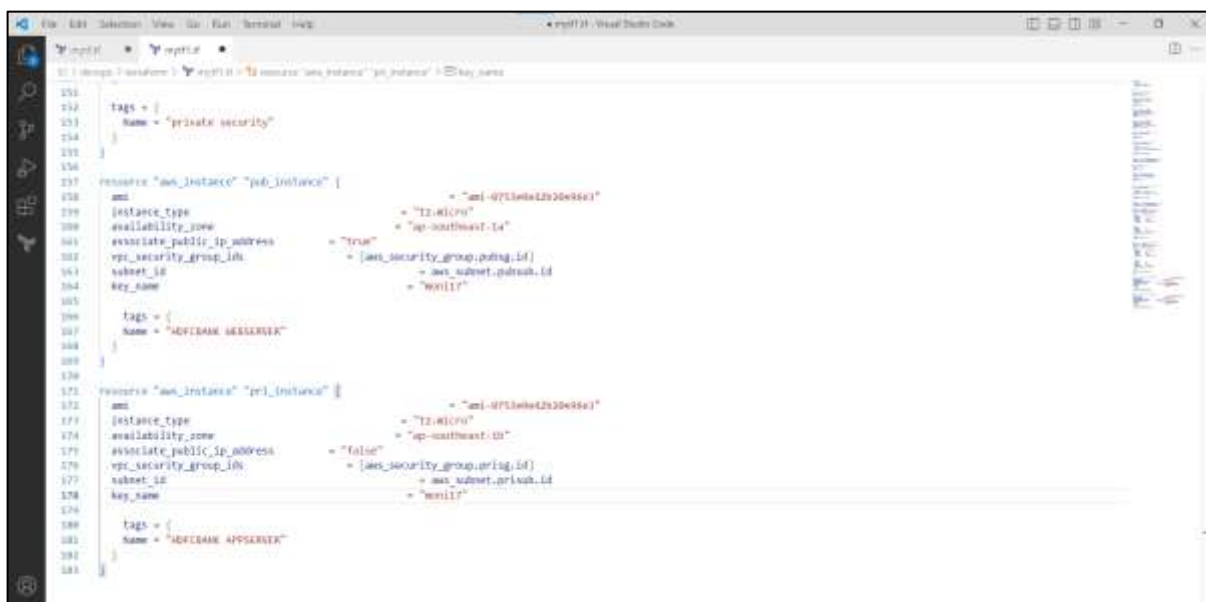
This code separte copy



## Step19: Write Code For private ec2 instance

This code separte copy

Same changes for public instance



## Step20:copy full code and put server

```
# cd /usr/local/bin
```

```
#vi myterraform.tf --->put code and save :wq!
```

[illegible]

Now initializing,

```
#terraform init
```

```
root@ip-172-31-23-109:~# cd /usr/local/bin
root@ip-172-31-23-109:/usr/local/bin# ll
total 5708
-rwxr-xr-x 2 root root 4096 Feb 24 05:21 ./
-rwxr-xr-x 10 root root 4096 Jan 11 13:49 ./
-rwxr-xr-x 1 root root 34084800 Nov 30 2021 terraform*
root@ip-172-31-23-109:/usr/local/bin# vi ~/.terraformrc
root@ip-172-31-23-109:/usr/local/bin# terraform init

initializing the backend...

initializing provider plugins...
- Floating hashicorp/aws versions matching ">= 3.0"...
- Installing hashicorp/aws v3.76.1...
- installed hashicorp/aws v3.76.1 (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 initialised!

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.
root@ip-172-31-23-109:/usr/local/bin#
```

Now terraform plan

#terraform plan

```
root@ip-172-31-23-109:/usr/local/bin# terraform plan
# aws_vpc.myvpc will be created
+ resource "aws_vpc" "myvpc" {
+   cidr_block = "10.0.0.0/16" = (known after apply)
+   default_network_acl_id = (known after apply)
+   default_route_table_id = (known after apply)
+   default_security_group_id = (known after apply)
+   dhcp_options_id = (known after apply)
+   enable_classiclink = (known after apply)
+   enable_classiclink_dns_support = (known after apply)
+   enable_dns_hostnames = (known after apply)
+   enable_dns_support = true = (known after apply)
+   id = (known after apply)
+   instance_tenancy = "default" = (known after apply)
+   ipv6_cidr_block = (known after apply)
+   ipv6_cidr_block_network_border_group = (known after apply)
+   main_route_table_id = (known after apply)
+   nat_id = (known after apply)
+   tags = {
+     "Name" = "My-VPC"
+   }
+   tags_all = {
+     "Name" = "My-VPC"
+   }
+ }

Plan: 14 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
```

Now apply terraform

#terraform apply ---->enter value:yes---->vpc,subnt,rt table.ec2 server,eip,nat,igw are create

```
root@ip-172-31-23-109:/usr/local/bin# terraform apply
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_vpc.myvpc: Creating...
aws_eip.teip: Creating...
aws_eip.teip: Creation complete after 1s [id=ipalloc-029446361bc346c2ff]
aws_vpc.myvpc: Creation complete after 1s [id=vpc-080ba00c03b0a3dd]
aws_internet_gateway.tigw: Creating...
aws_subnet.pr1sub: Creating...
aws_subnet.pubsub: Creating...
aws_security_group.psg: Creating...
aws_security_group.psg: Creation complete after 1s [id=sg-UTb5cf48529888c19]
aws_internet_gateway.tigw: Creation complete after 0s [id=igw-UTb5cf48529888c19]
aws_route_table.prtb: Creating...
aws_subnet.pr1sub: Creation complete after 1s [id=subnet-010fa06c7bf71cde]
aws_subnet.pubsub: Creation complete after 1s [id=subnet-041ed758c1ff6b2c2]
aws_nat_gateway.mynat: Creating...
aws_route_table.prtb: Creation complete after 0s [id=rtb-0b2277d3916b04ff1]
aws_route_table_association.prtasoc: Creating...
aws_route_table_association.prtasoc: Creation complete after 0s [id=rtbasoc-06c8b34f29b0b86dc]
aws_security_group.psg: Creation complete after 2s [id=sg-0d9864d87f6d64b6]
aws_instance.pr1instance: Creating...
aws_instance.pubinstance: Creating...
aws_nat_gateway.mynat: Still creating... [10s elapsed]
aws_instance.pr1instance: Still creating... [10s elapsed]
aws_instance.pubinstance: Still creating... [10s elapsed]
aws_nat_gateway.mynat: Still creating... [20s elapsed]
aws_instance.pr1instance: Still creating... [20s elapsed]
aws_instance.pubinstance: Still creating... [20s elapsed]
aws_nat_gateway.mynat: Still creating... [30s elapsed]
aws_instance.pr1instance: Still creating... [30s elapsed]
aws_instance.pubinstance: Still creating... [30s elapsed]
aws_nat_gateway.mynat: Still creating... [40s elapsed]
aws_instance.pr1instance: Still creating... [40s elapsed]
aws_instance.pubinstance: Still creating... [40s elapsed]
aws_nat_gateway.mynat: Still creating... [50s elapsed]
aws_instance.pr1instance: Still creating... [50s elapsed]
aws_instance.pubinstance: Still creating... [50s elapsed]
aws_nat_gateway.mynat: Still creating... [1m0s elapsed]
aws_instance.pr1instance: Still creating... [1m0s elapsed]
aws_instance.pubinstance: Still creating... [1m0s elapsed]
aws_nat_gateway.mynat: Still creating... [1m20s elapsed]
aws_instance.pr1instance: Still creating... [1m20s elapsed]
aws_instance.pubinstance: Still creating... [1m20s elapsed]
aws_nat_gateway.mynat: Creation complete after 1s22s [id=nat-08c4dd361c05828ea]
aws_route_table.prtb: Creation complete after 1s [id=rtb-0c9e583b3561f46b4]
aws_route_table_association.prtasoc: Creation complete after 0s [id=rtbasoc-0b6e1056fa3616f]
aws_route_table_association.prtasoc: Creation complete after 0s [id=rtbasoc-0b6e1056fa3616f]

Apply complete! Resources: 14 added, 0 changed, 0 destroyed.

root@ip-172-31-23-109:/usr/local/bin#
```

All are created.



Instances (EC2 Management Console) | ap-south-east-1.console.aws.amazon.com/ec2/home?region=ap-south-east-1#instances:tag:Name=HDFCBANK

Instances (1/3) info

Find instance by attribute or tag (name, version)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
terafarm	i-0a6a0173ea81d9e7	Running	t2.micro	2/2 checks passed	No alarms	ap-south-east-1a
HDFCBANK WEBSERVER	i-0b6bed00e95e58574	Running	t2.micro	2/2 checks passed	No alarms	ap-south-east-1a
HDFCBANK APPSERVER	i-0a6a0173ea81d9e7	Running	t2.micro	2/2 checks passed	No alarms	ap-south-east-1a

Instance: i-0b6bed00e95e58574 (HDFCBANK WEBSERVER)

Details | Security | Networking | Storage | Status checks | Monitoring | Tags

Instance summary info

Instance ID: i-0b6bed00e95e58574 (HDFCBANK WEBSERVER)

Public IPv4 address: 3.120.4 | open address

Private IPv4 addresses: 10.0.1.182

IPv4 address: Instance state: Public IPv4 DNS:

Instances (EC2 Management Console) | Your VPCs (VPC Management Console) | ap-south-east-1.console.aws.amazon.com/vpc/home?region=ap-south-east-1#vpcs

Your VPCs (2) info

Filter VPCs

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP
MY-VPC	vpc-0908a000f0d0e95d	Available	10.0.0.0/16	-	dhcp-G
-	vpc-0567e7549e0161560	Available	172.31.0.0/16	-	dhcp-G

Select a VPC above

Instances (EC2 Management Console) | Subnets (VPC Management Console) | ap-south-east-1.console.aws.amazon.com/vpc/home?region=ap-south-east-1#subnets

Subnets (5) info

Filter subnets

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
-	subnet-00a344d89e17994	Available	vpc-0567e7549e0161560	172.31.16.0/20	-
private subnet	subnet-018f6097b771cde	Available	vpc-0908a000f0d0e95d   MY-VPC	10.0.2.0/24	-
-	subnet-007e6e0526abd0c33	Available	vpc-0567e7549e0161560	172.31.0.0/20	-
public subnet	subnet-043e8758c9ff6bc2	Available	vpc-0908a000f0d0e95d   MY-VPC	10.0.1.0/24	-
-	subnet-01d5d95c6424e088	Available	vpc-0567e7549e0161560	172.31.32.0/20	-

## Step21:Now Connected Public Instance

#vi mumbaikey --->put .pem

```
root@ip-172-31-23-109:~# vi mumbaikey
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA...
-----END RSA PRIVATE KEY-----
```

# chmod 400 mumbaikey

#ssh -i mumbaikey [ec2-user@3.1.204.4](https://ec2-user@3.1.204.4) --->yes

Open public server

```
ec2-user@ip-10-0-1-182:~
root@ip-172-31-23-109:/usr/local/bin# vi mumbaikey
root@ip-172-31-23-109:/usr/local/bin# chmod 400 mumbaikey
root@ip-172-31-23-109:/usr/local/bin# ssh -i mumbaikey ec2-user@3.1.204.4
The authenticity of host '3.1.204.4 (3.1.204.4)' can't be established.
ECDSA key fingerprint is SHA256:FuzkFVrytyrf3QMfUXGdDtcAvVmcie8JUrbeAte52GjY.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '3.1.204.4' (ECDSA) to the list of known hosts.

 _ _ | _ _ |
 _ _ | _ _ | / Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
16 package(s) needed for security, out of 16 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-0-1-182 ~]$ ^C
[ec2-user@ip-10-0-1-182 ~]$
```

Now check internet

#ping google.com



```

new you will pay when an instance terminates (yes/no)? no
Warning: Permanently added '3.1.234.4' (ECDSA) to the list of known hosts.

  ____  _
 / ___|| | | |
| |___| | | |
 \___|| | | |
      |_| |_|

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
16 package(s) needed for security, out of 16 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-0-1-182 ~]$ ^C
[ec2-user@ip-10-0-1-182 ~]$ ping google.com
PING google.com (142.251.10.100) 56(84) bytes of data:
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=1 ttl=104 time=1.46 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=2 ttl=104 time=1.54 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=3 ttl=104 time=1.50 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=4 ttl=104 time=1.51 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=5 ttl=104 time=1.53 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=6 ttl=104 time=1.48 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=7 ttl=104 time=1.50 ms
^C
--- google.com ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6010ms
rtt min/avg/max/mdev = 1.468/1.506/1.540/0.048 ms
[ec2-user@ip-10-0-1-182 ~]$

```

Packets show so internet is working..

## Step22:Now Connected Private Instance

#vi mumbaikey --->put .pem

# chmod 400 mumbaikey

#ssh -i mumbaikey ec2-user@10.0.2.144 --->yes

Open private server

Check internt --->ping google.com

```

PING google.com (142.251.10.100) 56(84) bytes of data:
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=1 ttl=104 time=1.46 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=2 ttl=104 time=1.54 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=3 ttl=104 time=1.50 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=4 ttl=104 time=1.51 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=5 ttl=104 time=1.53 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=6 ttl=104 time=1.48 ms
64 bytes from sd-in-f100.1e100.net (142.251.10.100): icmp_seq=7 ttl=104 time=1.50 ms
^C
--- google.com ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6010ms
rtt min/avg/max/mdev = 1.468/1.506/1.540/0.048 ms
[ec2-user@ip-10-0-1-182 ~]$ vi mumbaikey
[ec2-user@ip-10-0-1-182 ~]$ chmod 400 mumbaikey
[ec2-user@ip-10-0-1-182 ~]$ ssh -i mumbaikey ec2-user@10.0.2.144
The authenticity of host '10.0.2.144 (10.0.2.144)' can't be established.
ECDSA key fingerprint is SHA256:JGeth7BeRaoXnyKilCYHaw47rNillYqs4b80FVNRzXyY.
ECDSA key fingerprint is MD5:64:9b:71:53:98:27:0a:e1:95:98:a9:fc:65:92:f0:df.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.144' (ECDSA) to the list of known hosts.

  ____  _
 / ___|| | | |
| |___| | | |
 \___|| | | |
      |_| |_|

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-2-144 ~]$ ping google.com
PING google.com (172.217.194.101) 56(84) bytes of data:
64 bytes from si-in-f101.1e100.net (172.217.194.101): icmp_seq=1 ttl=100 time=2.74 ms
64 bytes from si-in-f101.1e100.net (172.217.194.101): icmp_seq=2 ttl=100 time=2.33 ms
64 bytes from si-in-f101.1e100.net (172.217.194.101): icmp_seq=3 ttl=100 time=2.36 ms
64 bytes from si-in-f101.1e100.net (172.217.194.101): icmp_seq=4 ttl=100 time=2.29 ms
64 bytes from si-in-f101.1e100.net (172.217.194.101): icmp_seq=5 ttl=100 time=2.49 ms
^C
--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 5007ms
rtt min/avg/max/mdev = 2.296/2.432/2.741/0.163 ms
[ec2-user@ip-10-0-2-144 ~]$

```

## Step23:Delete All

#exit

#exit

#erroform destroy --->yes

```
root@ip-172-31-23-108:/usr/local/bin#
Do you really want to destroy all resources?
Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

aws_instance.pub_instance: Destroying... [id=i-3b3bed30a85e55574]
aws_route_table_association.pubrtasoc: Destroying... [id=rtaasoc-06c8b34f29c0b6fdb]
aws_route_table_association.prietasoc: Destroying... [id=rtaasoc-0c8e1103bfea2638f]
aws_instance.pri_instance: Destroying... [id=i-3dbbfb9771c8401cc]
aws_route_table_association.prietasoc: Destruction complete after 1s
aws_route_table.pri: Destroying... [id=rth-0cee589635e5f68b4]
aws_route_table_association.pubrtasoc: Destruction complete after 1s
aws_route_table.pubrt: Destroying... [id=rth-0b227b49916904ff1]
aws_route_table.pubrt: Destruction complete after 0s
aws_internet_gateway.tigw: Destroying... [id=igw-07b5cf40529909c19]
aws_route_table.pri: Destruction complete after 0s
aws_nat_gateway.mgnat: Destroying... [id=nat-09c4dd361c05820ea]
aws_instance.pub_instance: Still destroying... [id=i-3b3bed30a85e55574, 10s elapsed]
aws_instance.pri_instance: Still destroying... [id=i-3dbbfb9771c8401cc, 10s elapsed]
aws_internet_gateway.tigw: Still destroying... [id=igw-07b5cf40529909c19, 10s elapsed]
aws_nat_gateway.mgnat: Still destroying... [id=nat-09c4dd361c05820ea, 10s elapsed]
aws_instance.pub_instance: Still destroying... [id=i-3b3bed30a85e55574, 20s elapsed]
aws_instance.pri_instance: Still destroying... [id=i-3dbbfb9771c8401cc, 20s elapsed]
aws_internet_gateway.tigw: Still destroying... [id=igw-07b5cf40529909c19, 20s elapsed]
aws_nat_gateway.mgnat: Still destroying... [id=nat-09c4dd361c05820ea, 20s elapsed]
aws_instance.pub_instance: Still destroying... [id=i-3b3bed30a85e55574, 30s elapsed]
aws_instance.pri_instance: Still destroying... [id=i-3dbbfb9771c8401cc, 30s elapsed]
aws_internet_gateway.tigw: Still destroying... [id=igw-07b5cf40529909c19, 30s elapsed]
aws_nat_gateway.mgnat: Still destroying... [id=nat-09c4dd361c05820ea, 30s elapsed]
aws_internet_gateway.tigw: Destruction complete after 30s
aws_instance.pri_instance: Destruction complete after 40s
aws_security_group.pri_sg: Destroying... [id=sg-0d5864487fe3d4b68]
aws_subnet.priash: Destroying... [id=subnet-038fe609c7b7f1cde1]
aws_instance.pub_instance: Destruction complete after 40s
aws_security_group.pub_sg: Destroying... [id=sg-88f1c438a5495518]
aws_subnet.priash: Destruction complete after 0s
aws_security_group.pub_sg: Destruction complete after 1s
aws_security_group.pri_sg: Destruction complete after 1s
aws_nat_gateway.mgnat: Still destroying... [id=nat-09c4dd361c05820ea, 40s elapsed]
aws_nat_gateway.mgnat: Destruction complete after 40s
aws_elb.tel: Destroying... [id=elb-alpc-029444233bc28bc2f]
aws_subnet.pubash: Destroying... [id=subnet-043ed359cfff6dbca]
aws_subnet.priash: Destruction complete after 1s
aws_vpc.myvpc: Destroying... [id=vpc-0989a00f6d09eb5d]
aws_elb.tel: Destruction complete after 1s
aws_vpc.myvpc: Destruction complete after 0s

Destroy complete! Resources: 14 destroyed.
root@ip-172-31-23-108:/usr/local/bin#
```

After destroy iam role delete.