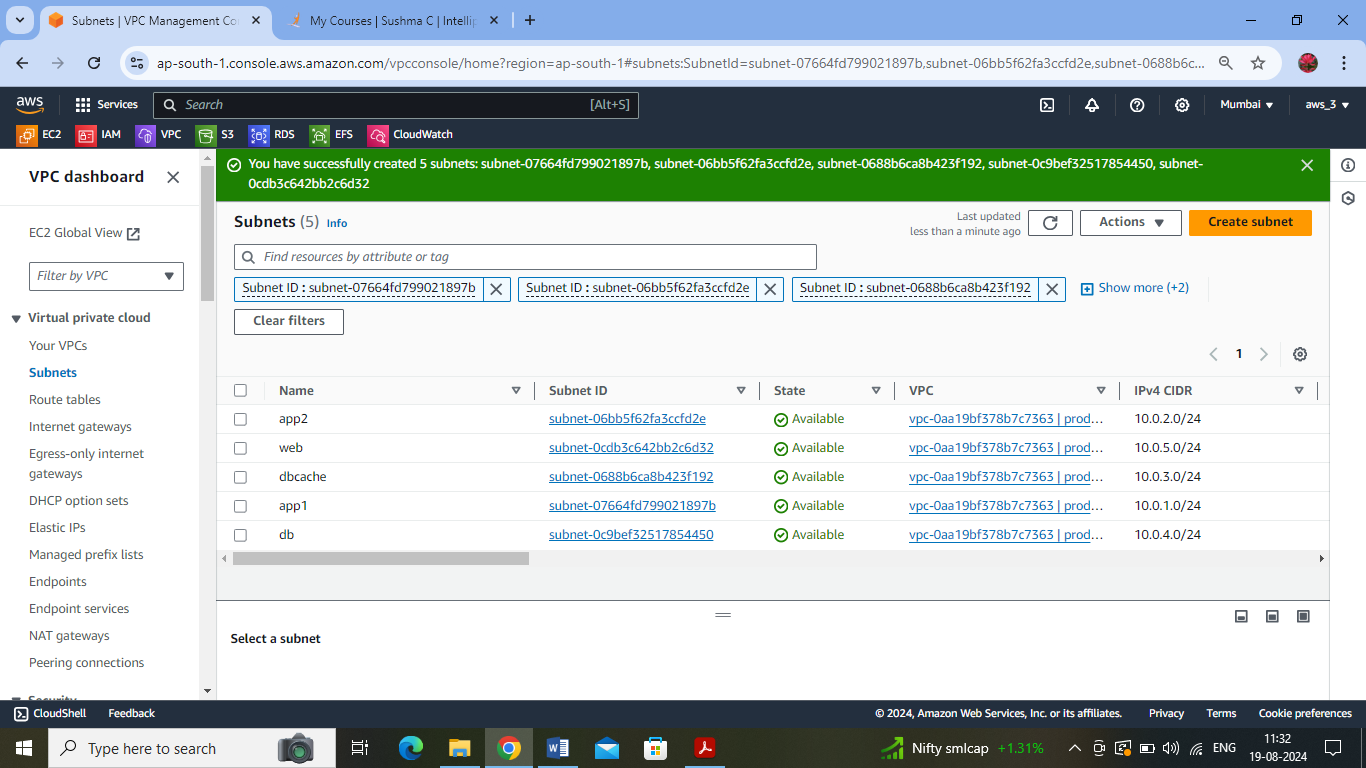
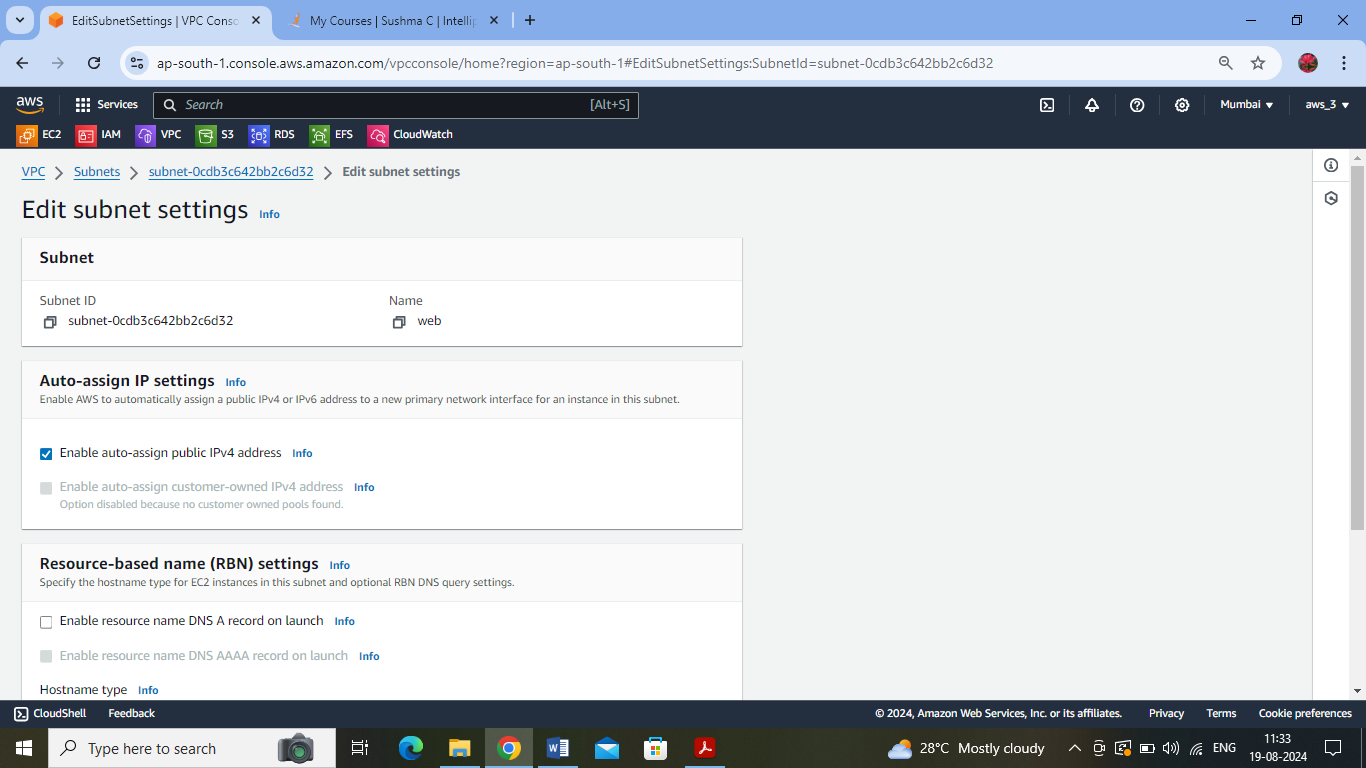
**Case Study : VPC**

Set the respective CIDRs, and give no preference on AZs in which VPC have to be created.

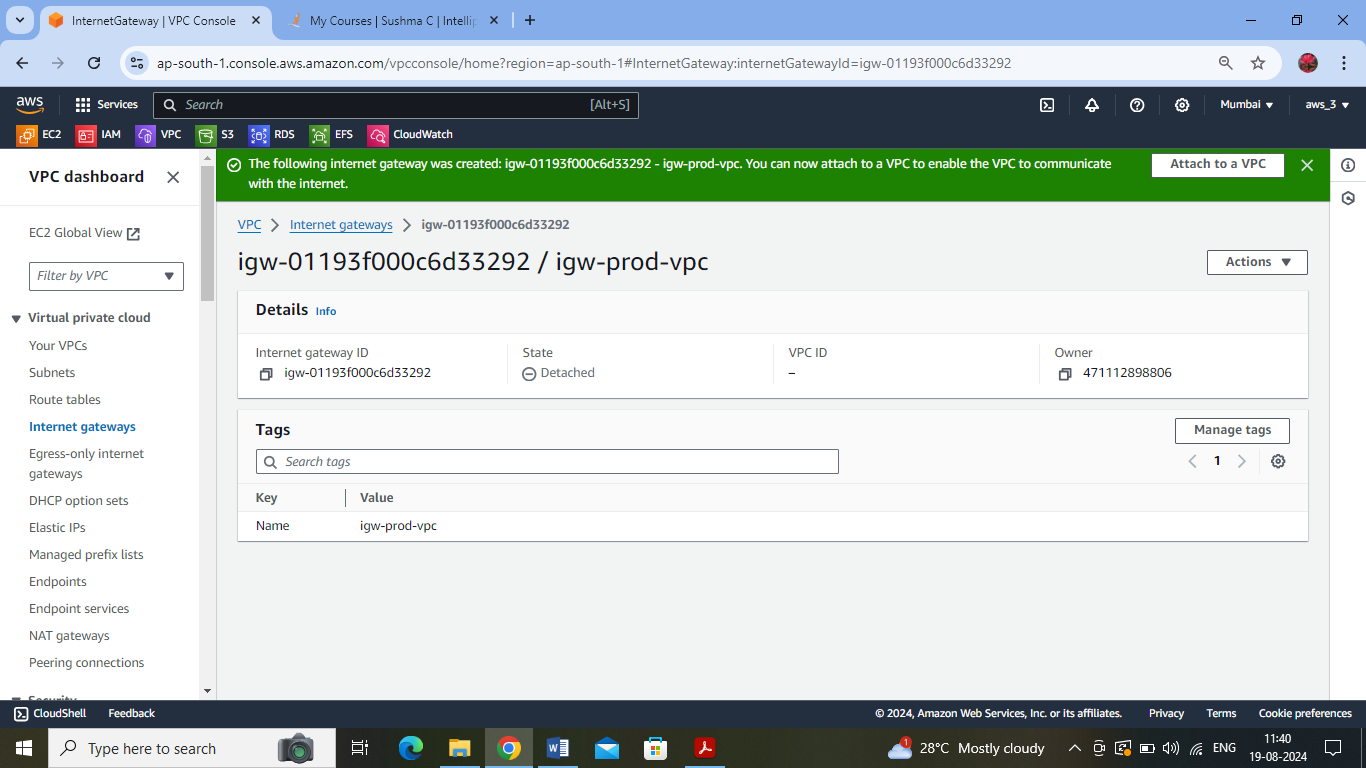
Creating the five subnets according to the 4-tier architecture suggested for the Production Network:



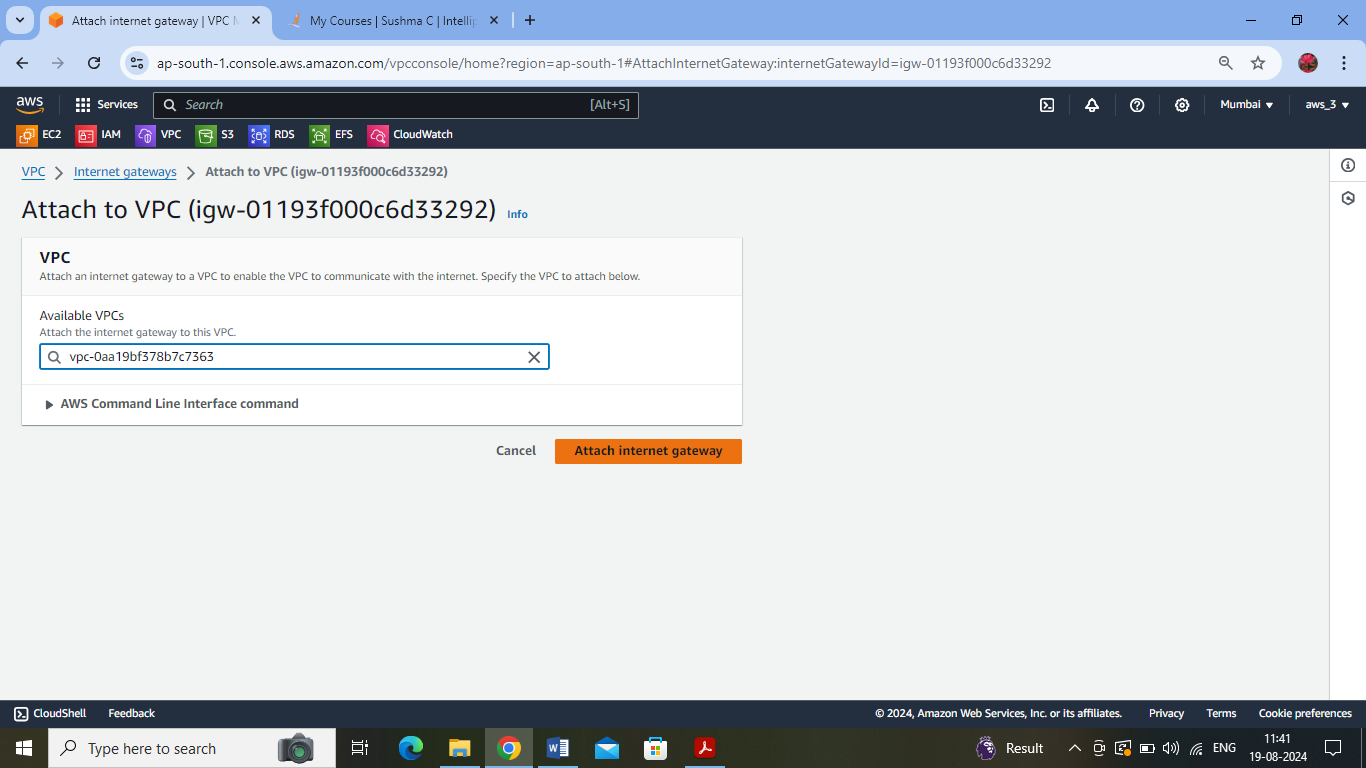
Assigning public ip address for the web subnet:



Created the IGW for the Production network :

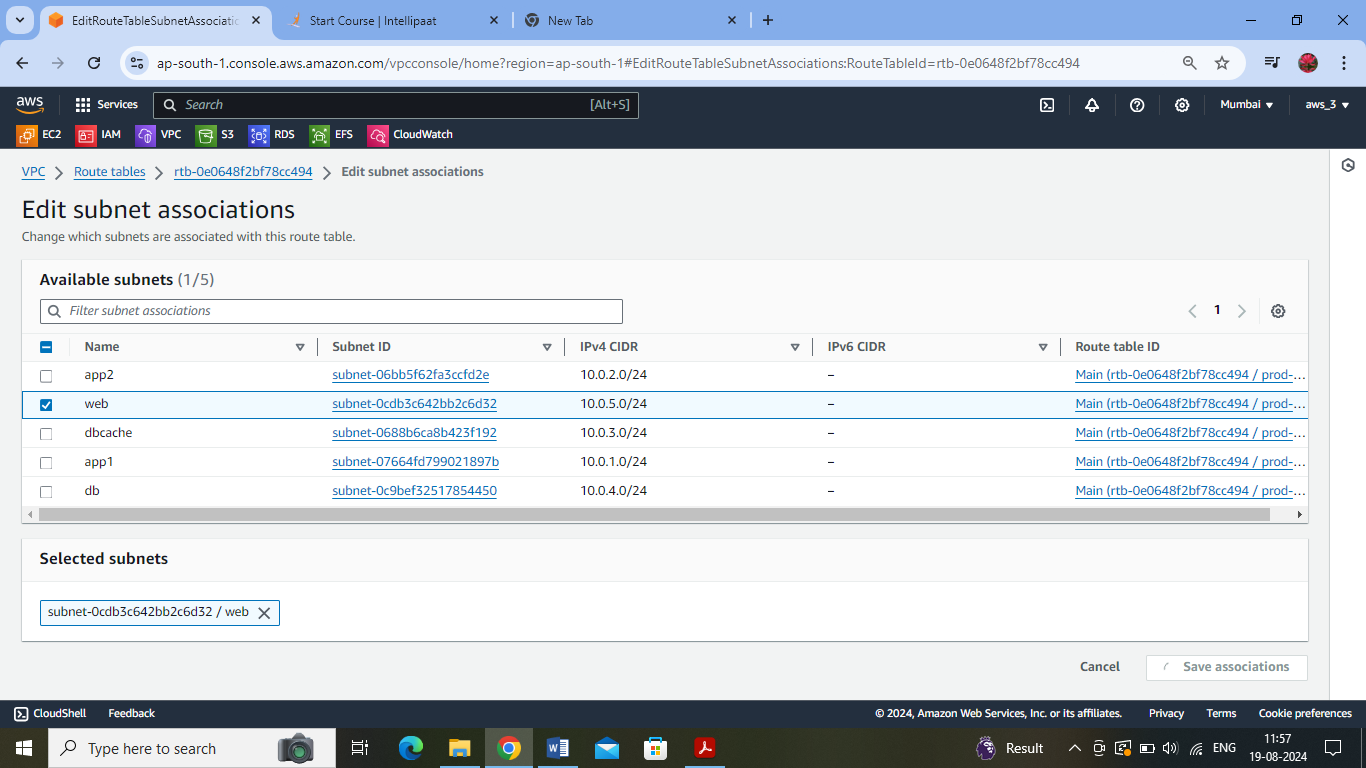


Attaching the IGW to the Production VPC:



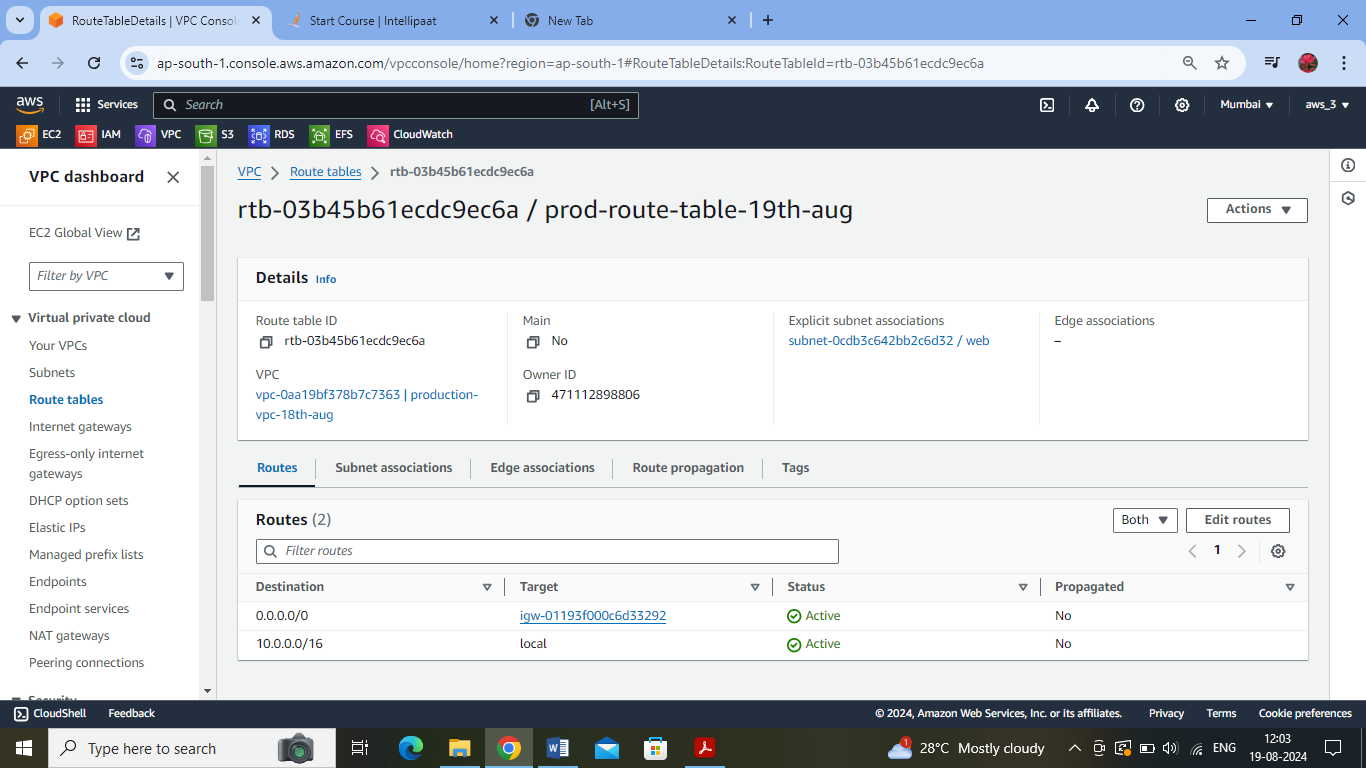
Adding the route to the Internet: destination 0.0.0.0/0 and target as the above created IGW:

Creating a subnet association between the web-subnet and the route table:



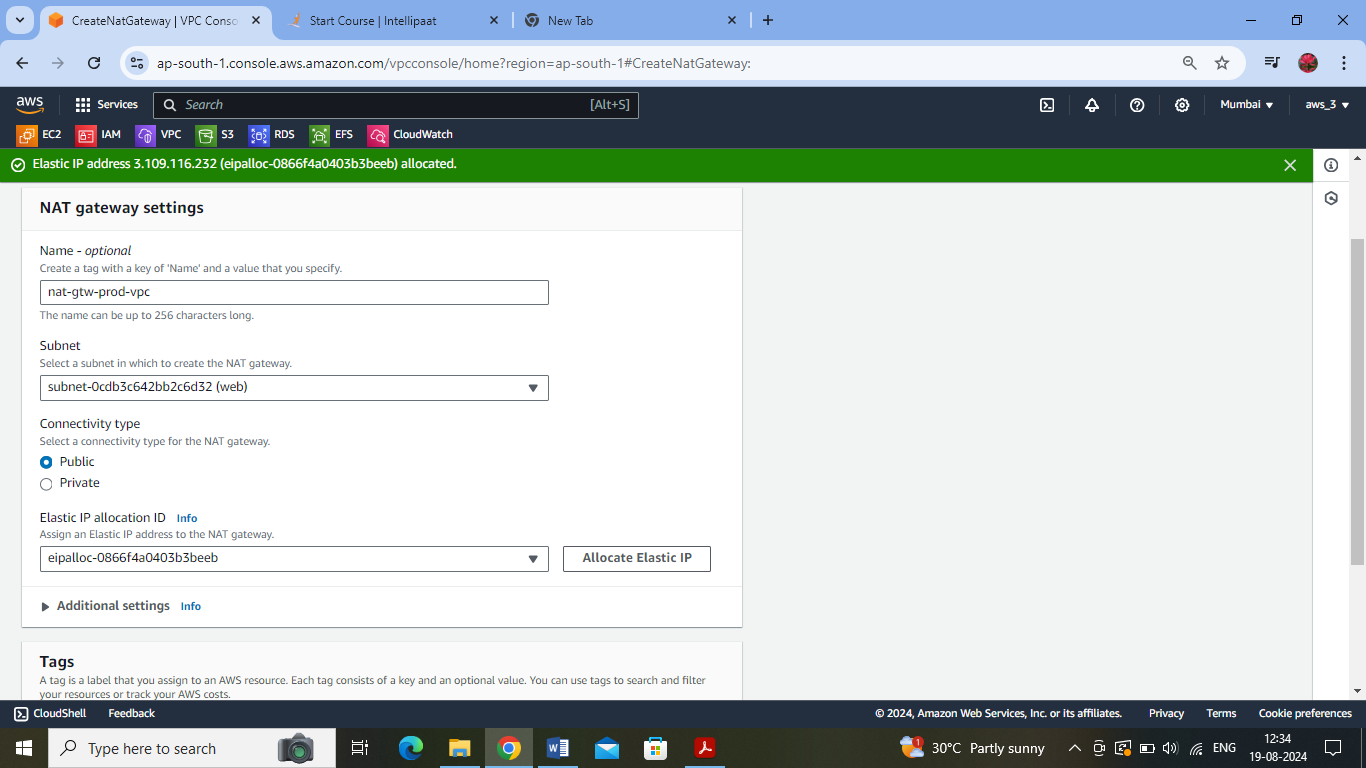
Creating a new route table (although a default route table is automatically created for our given vpc):

Added the route to the internet by attaching igw.

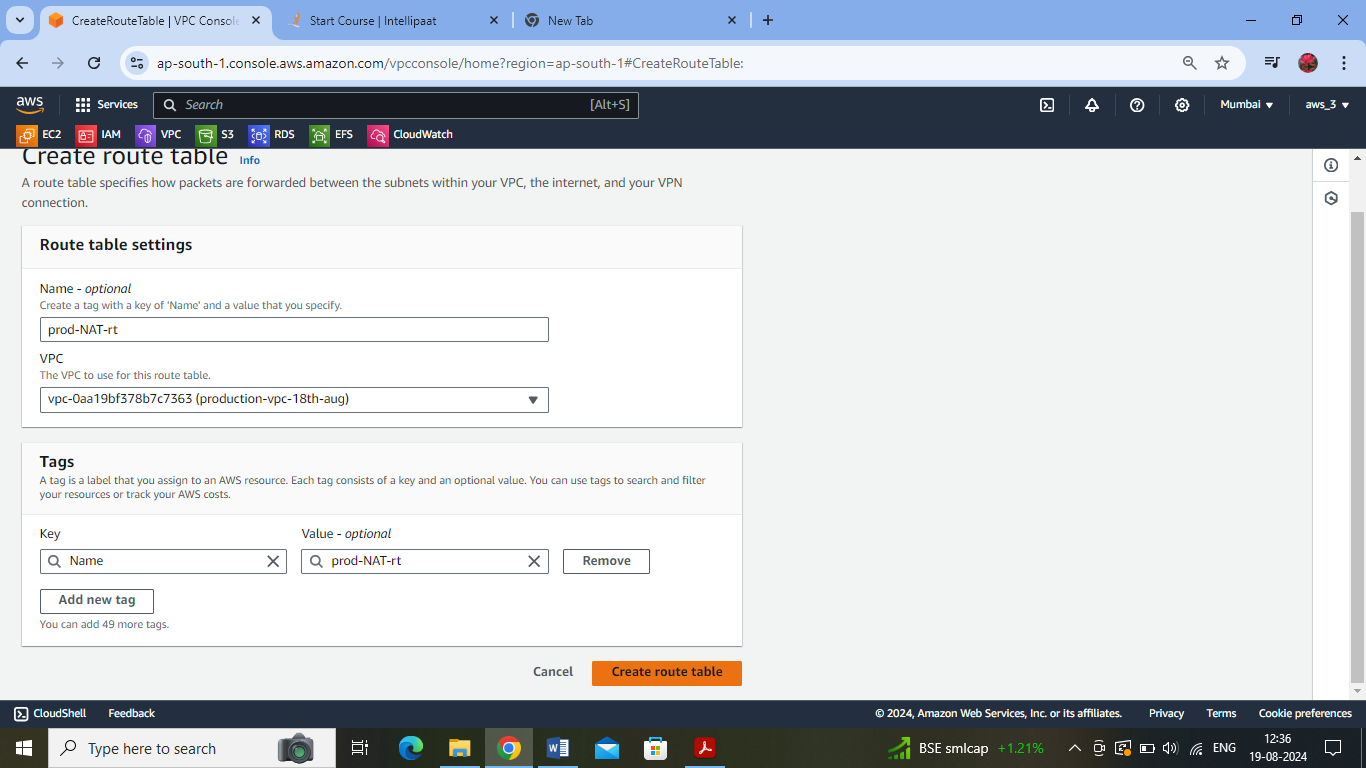


Associated this route table with the specific web subnet (as it is the only public subnet) :

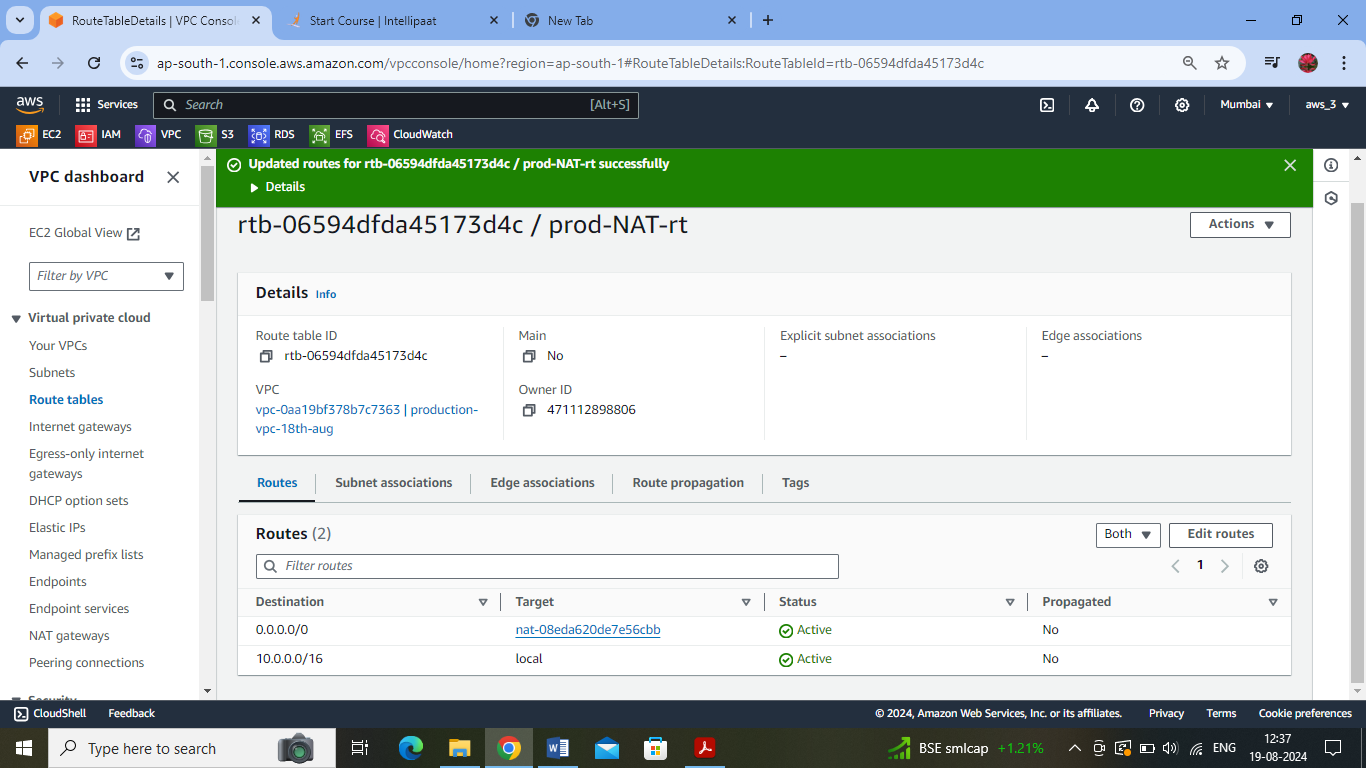
Creating the NAT gateway :



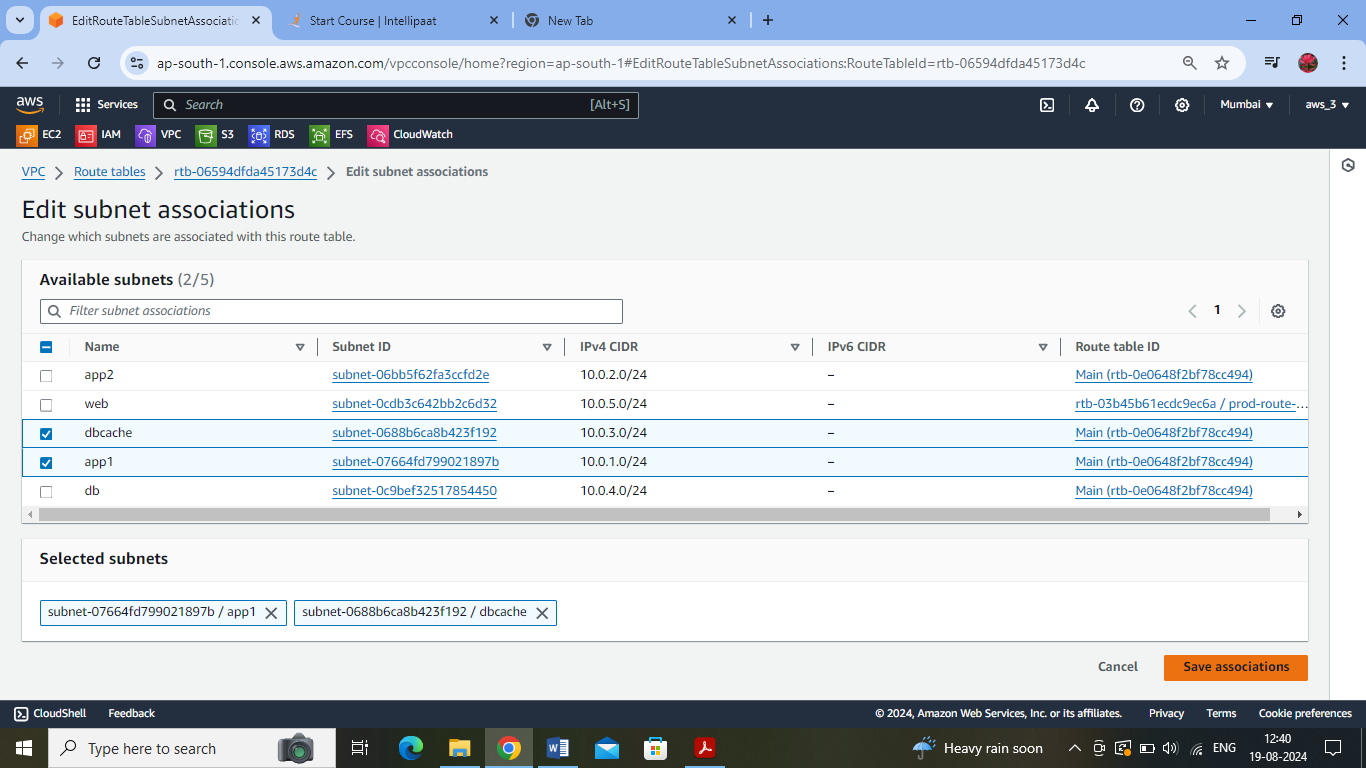
Creating a RT to associate the NAT gateway:



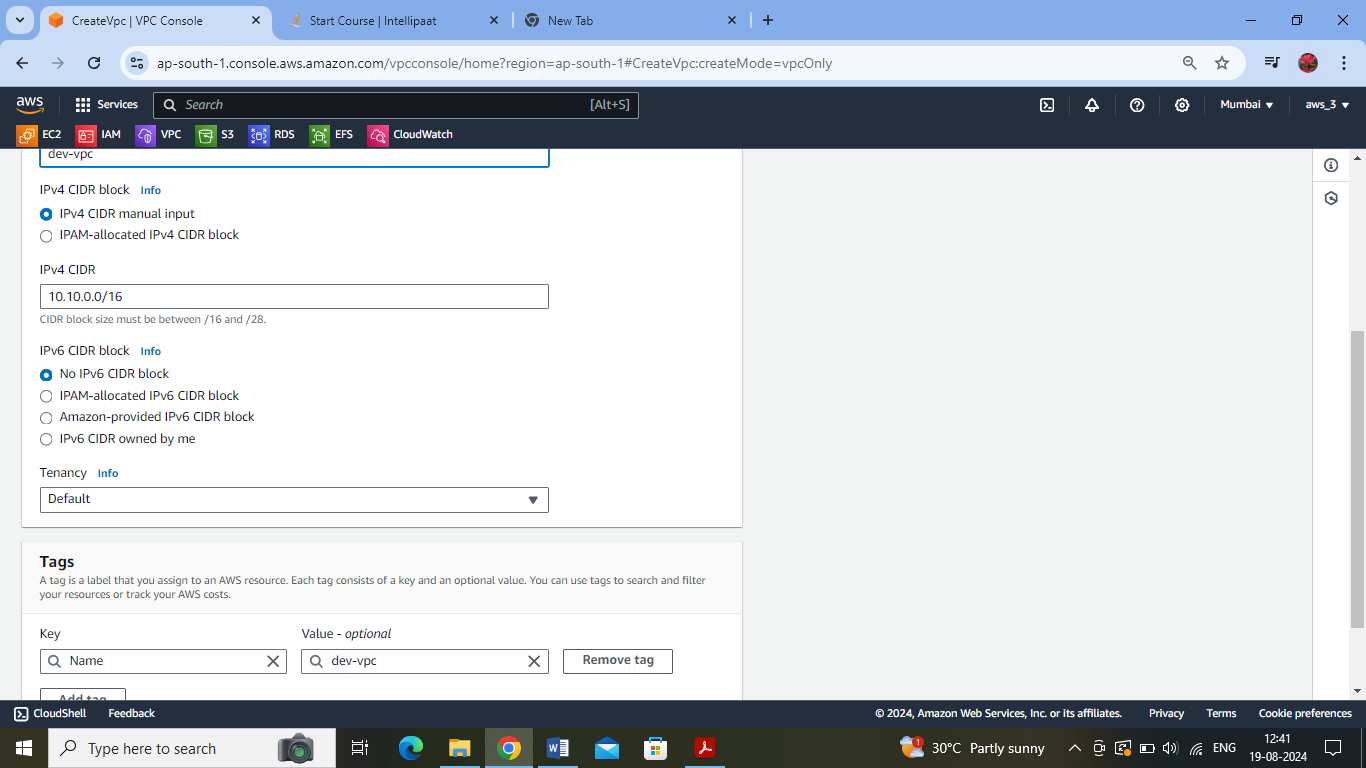
Added the destination of internet with target as NAT :



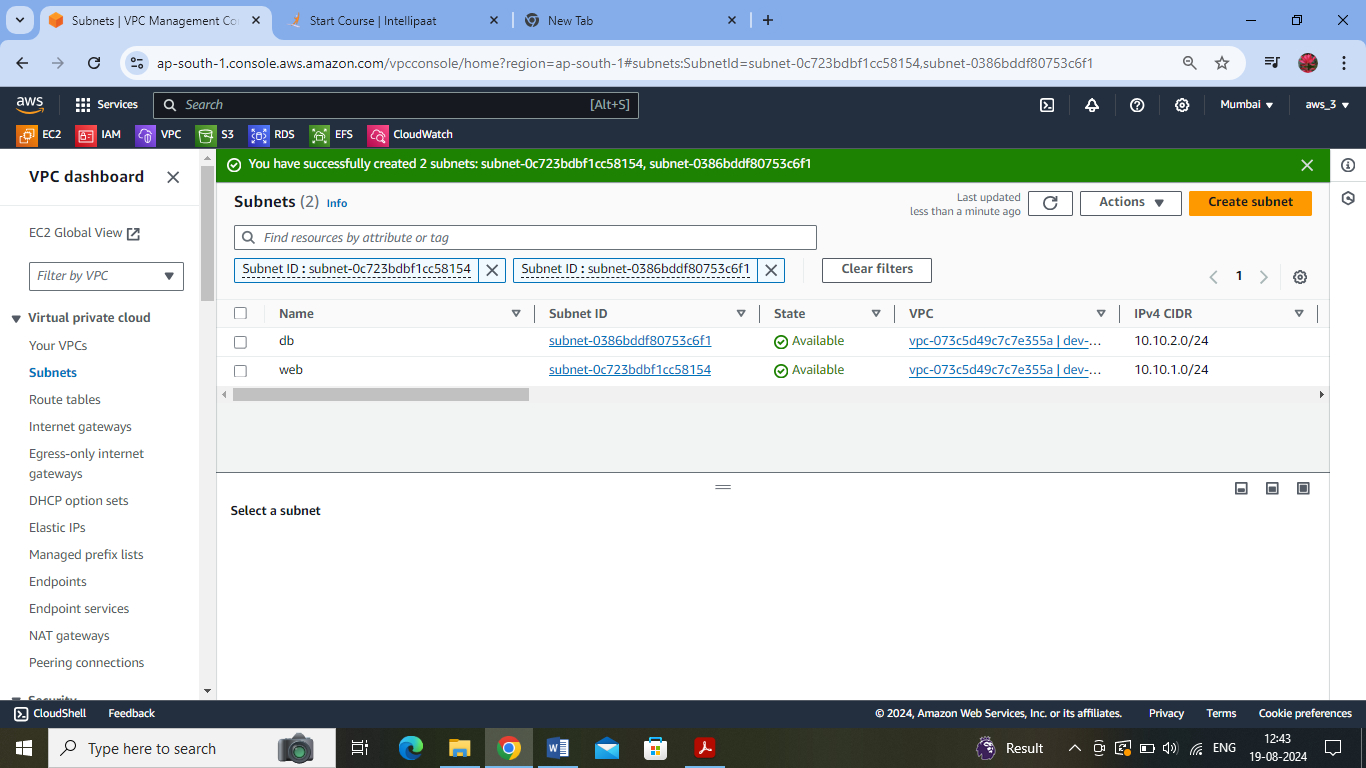
Associated the subnets app1 and dbcache to the RT:



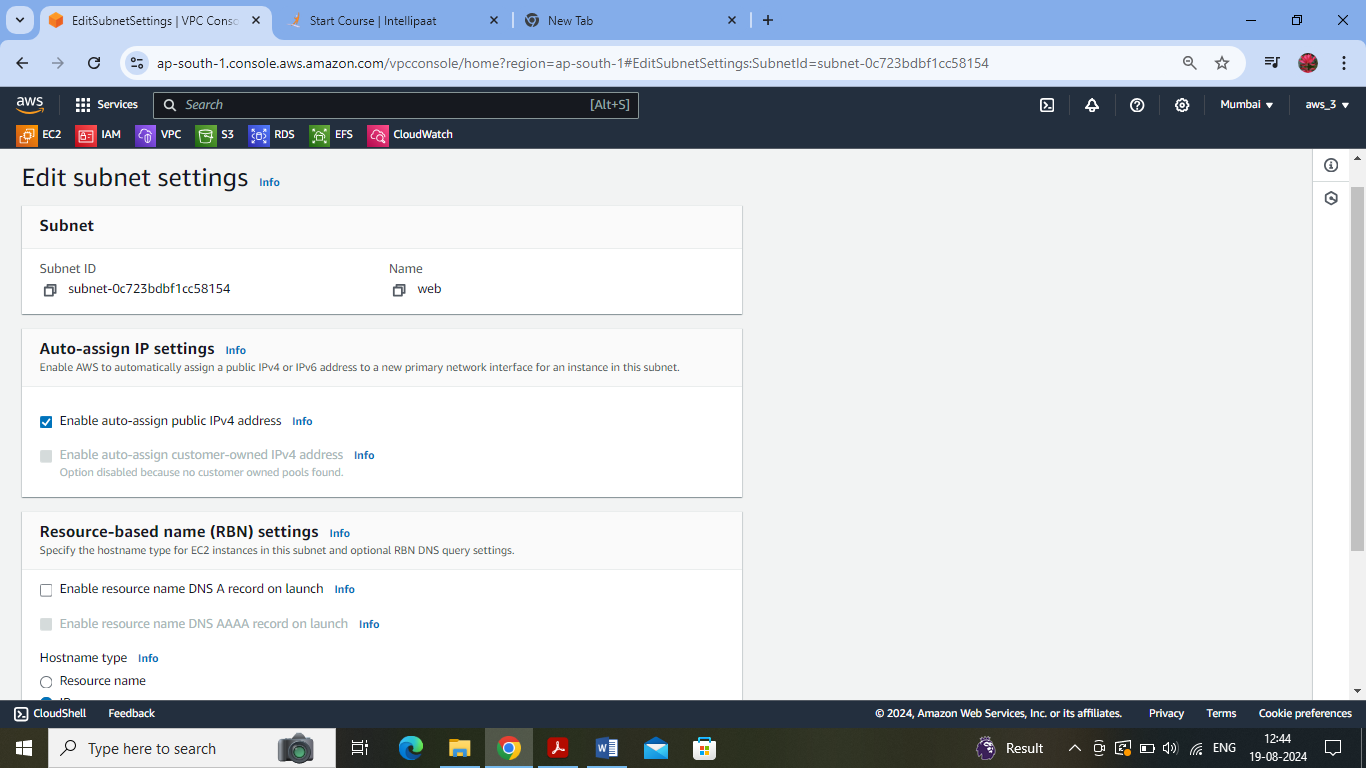
Created the 2-tier architecture of the Production Network :



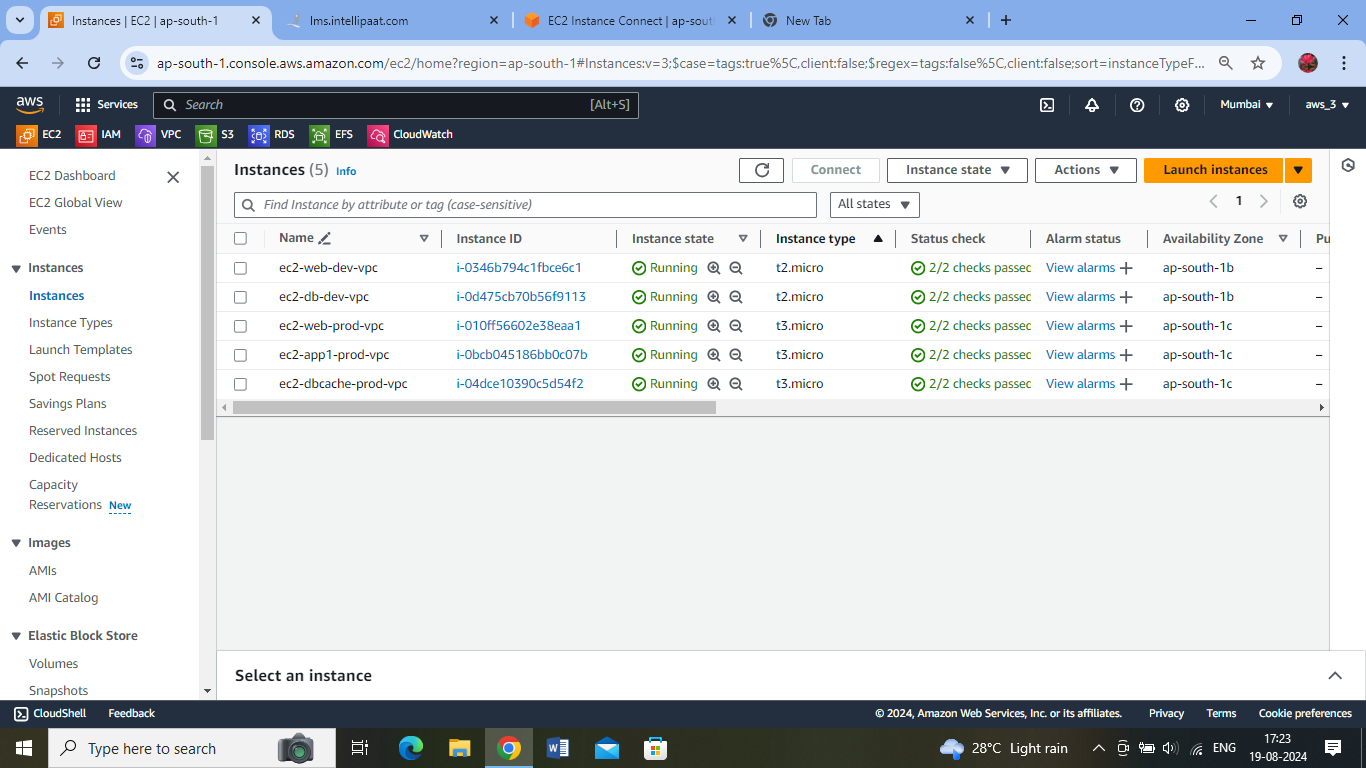
Db and web subnets



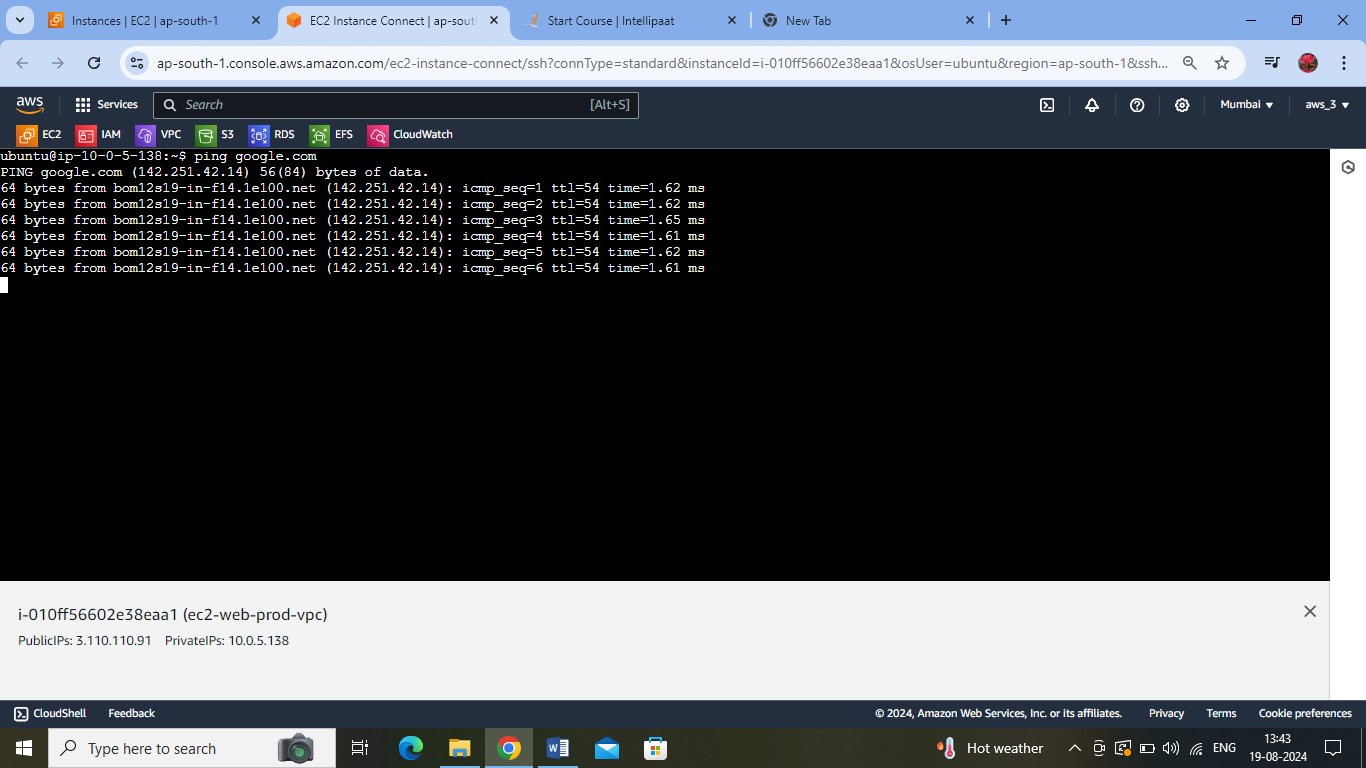
Web subnet is made a public subnet :



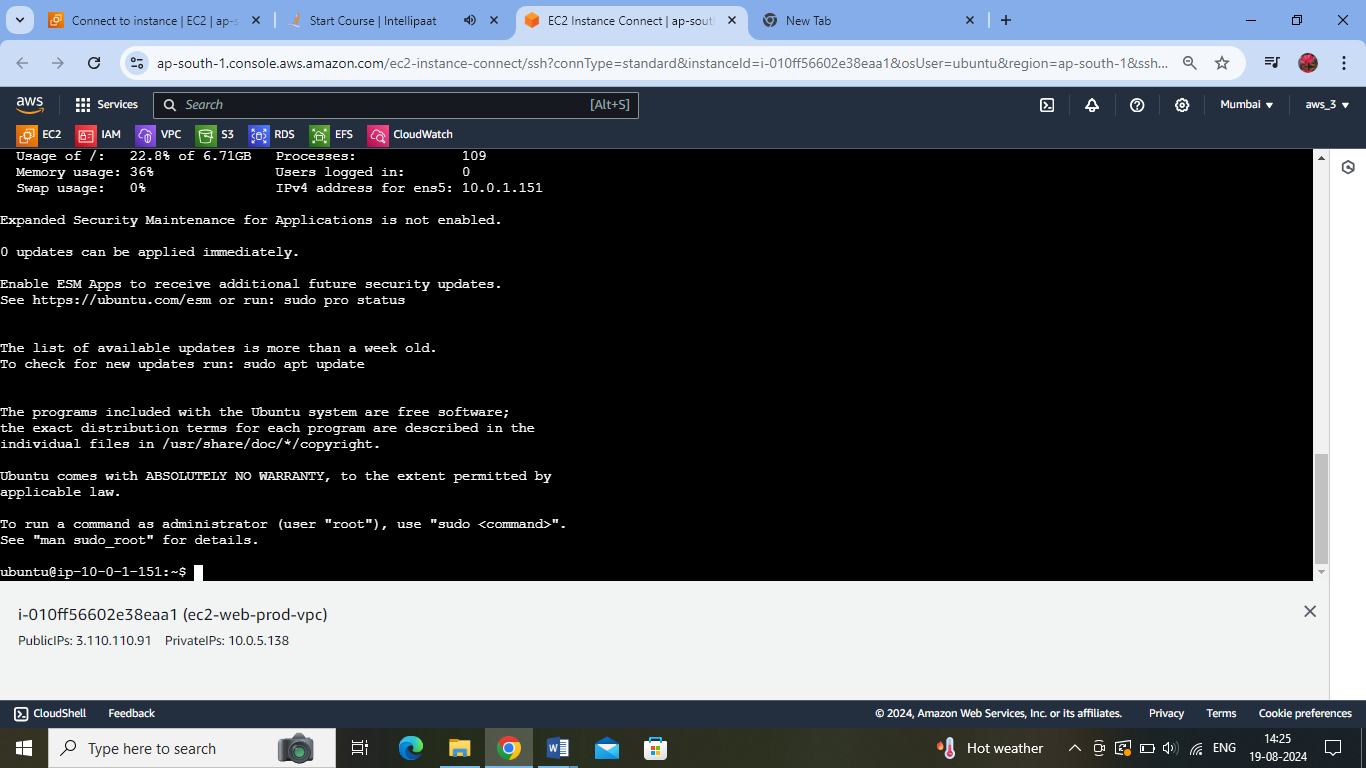
Created the respective instances in the respective subnets of each VPC :



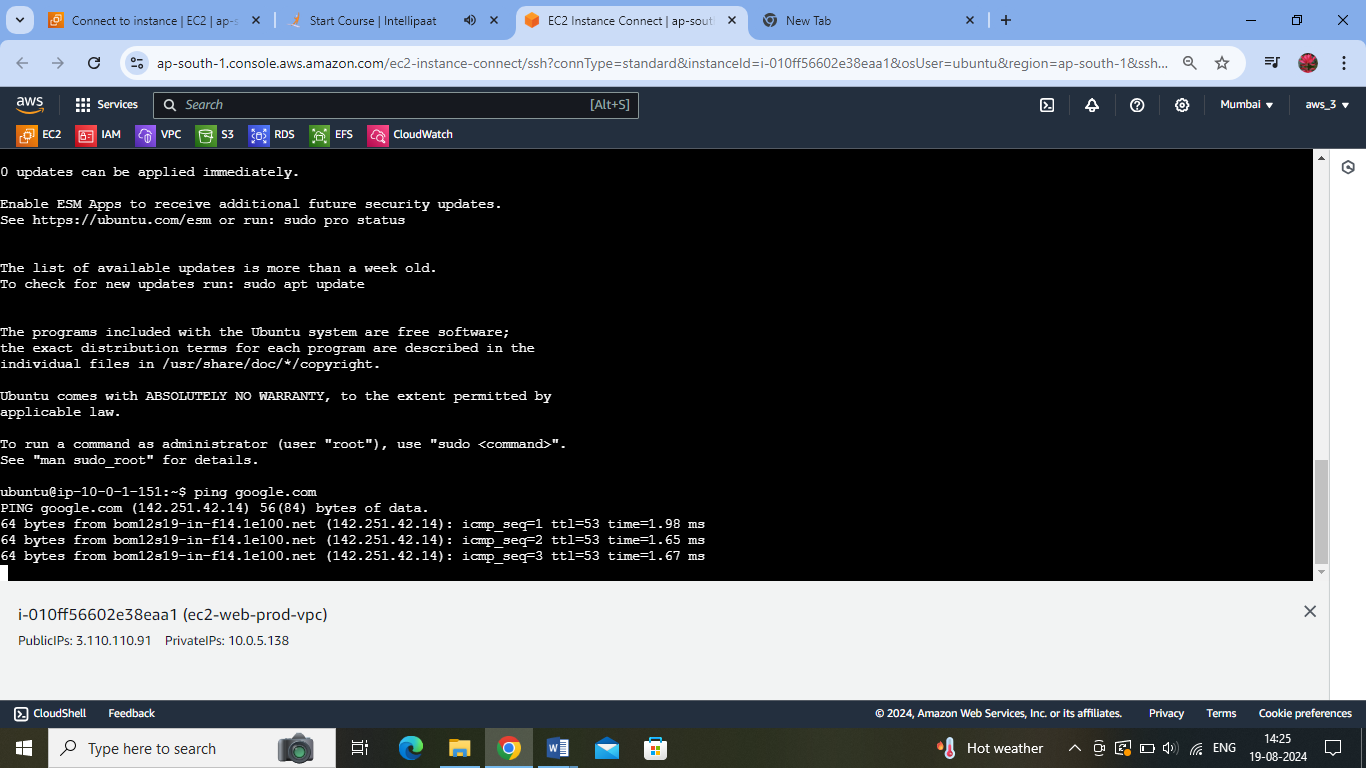
Connecting to ec2 instance – in web-prod-vpc : Able to access internet from web subnet of production network:



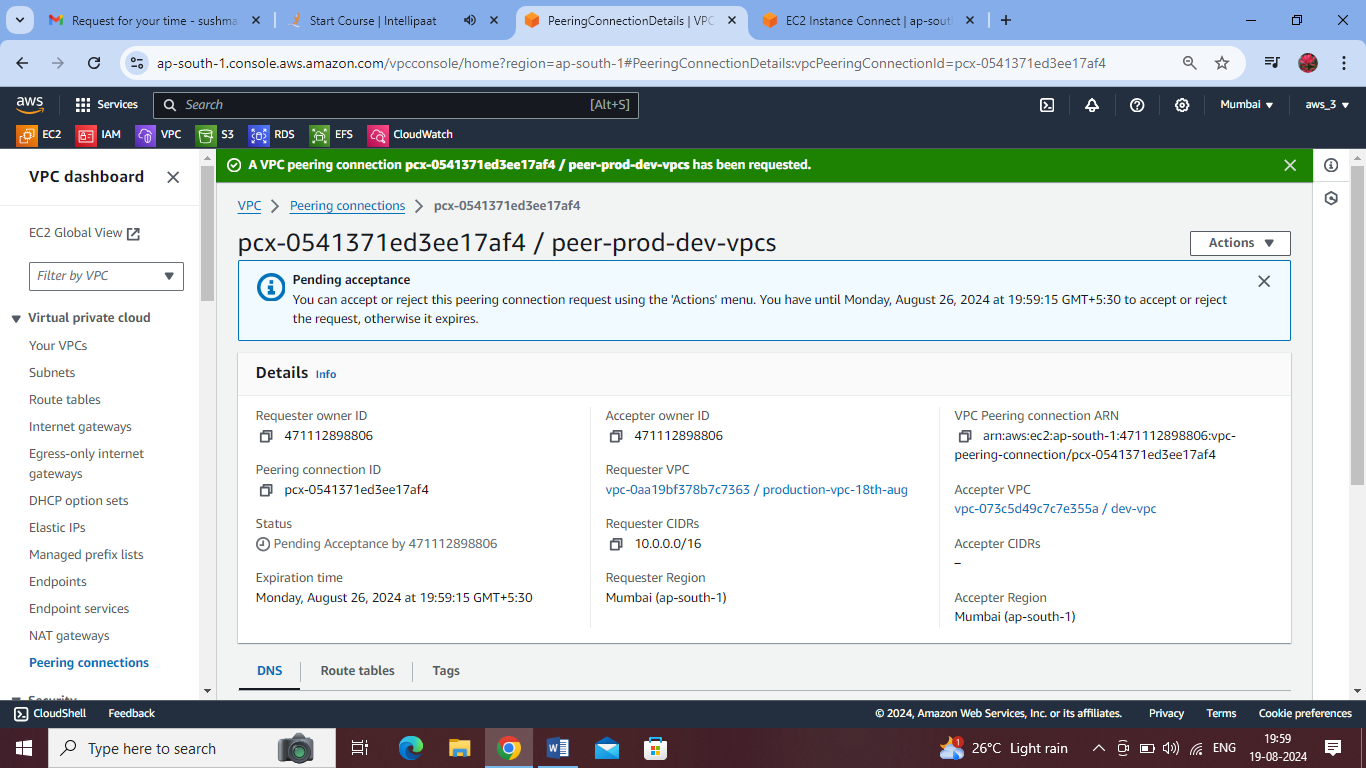
Logged into EC2 on app1 subnet using the Ec2 instance on web subnet:



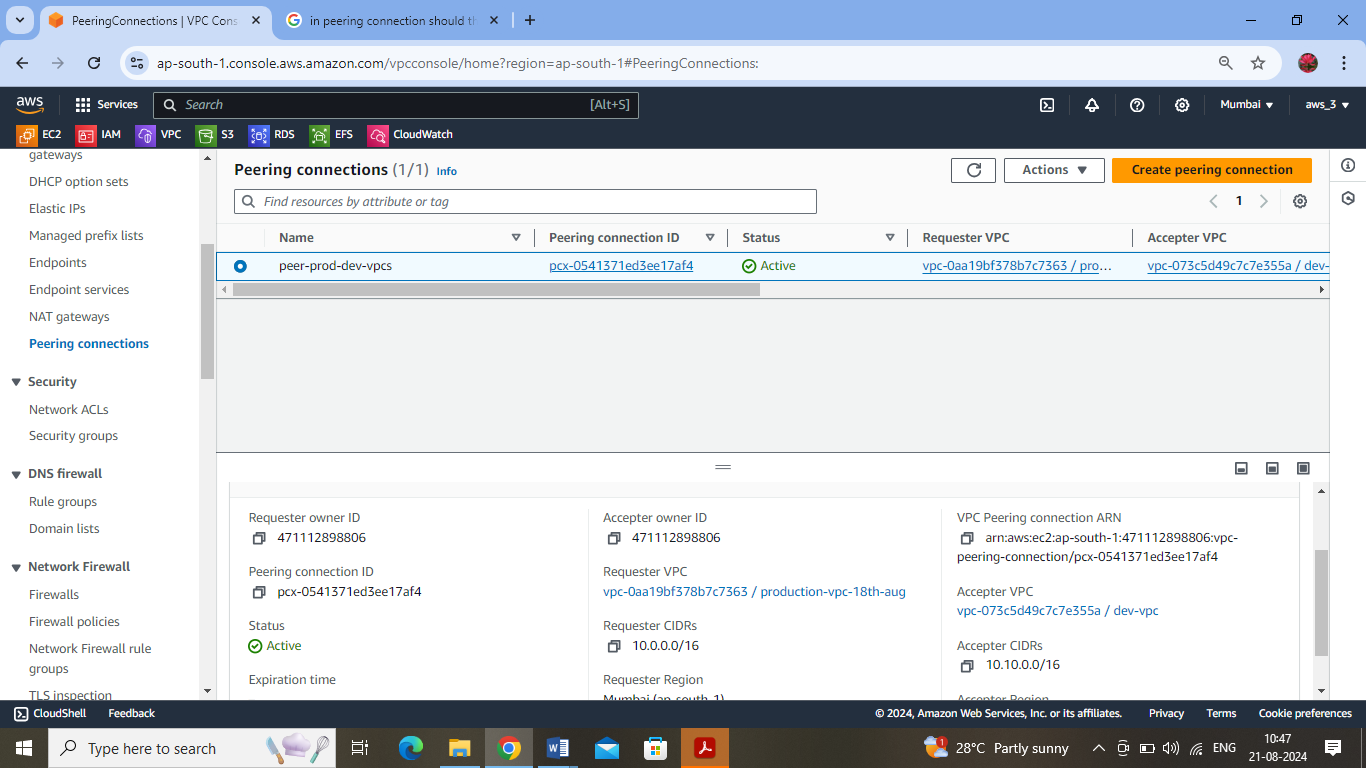
Able to access google from here as well. Hence the private subnet is able to access the internet via NAT gateway.



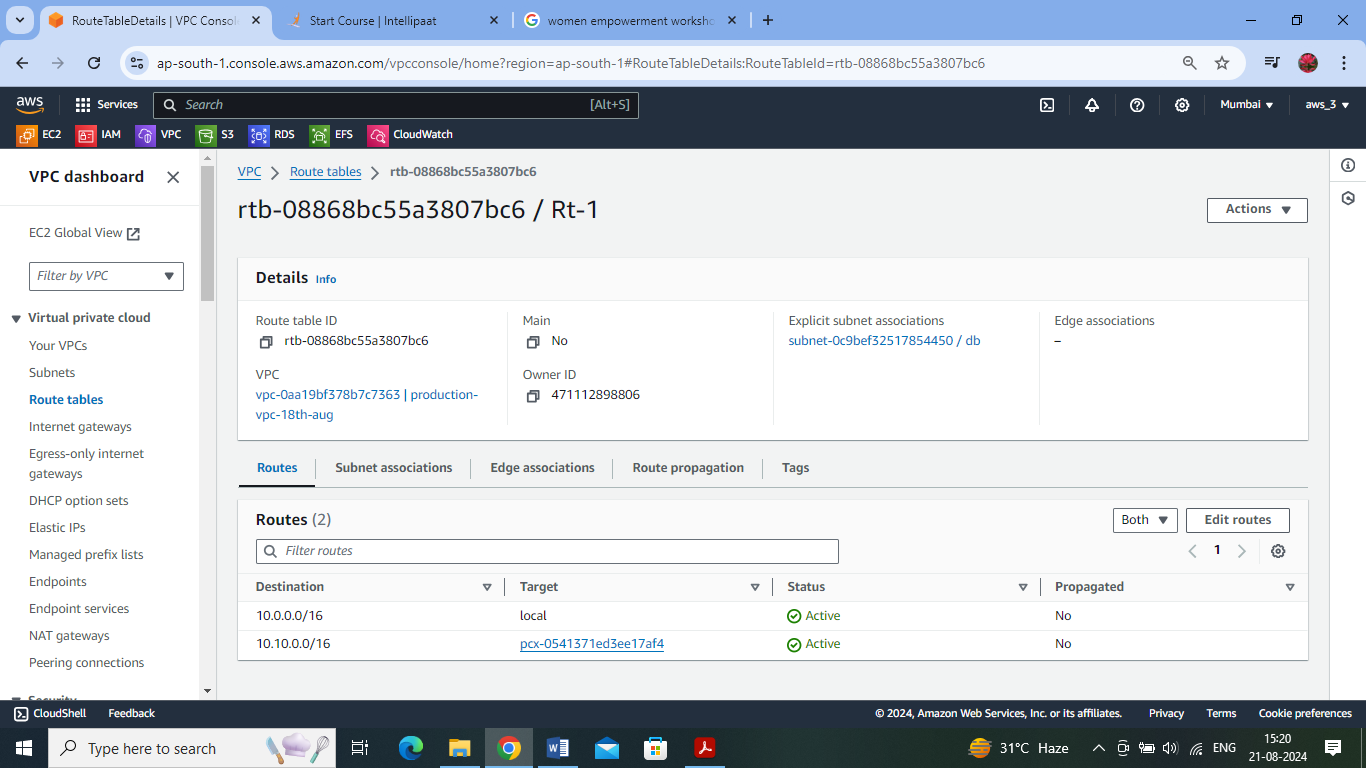
Created a Peering Connection between both the VPCs :



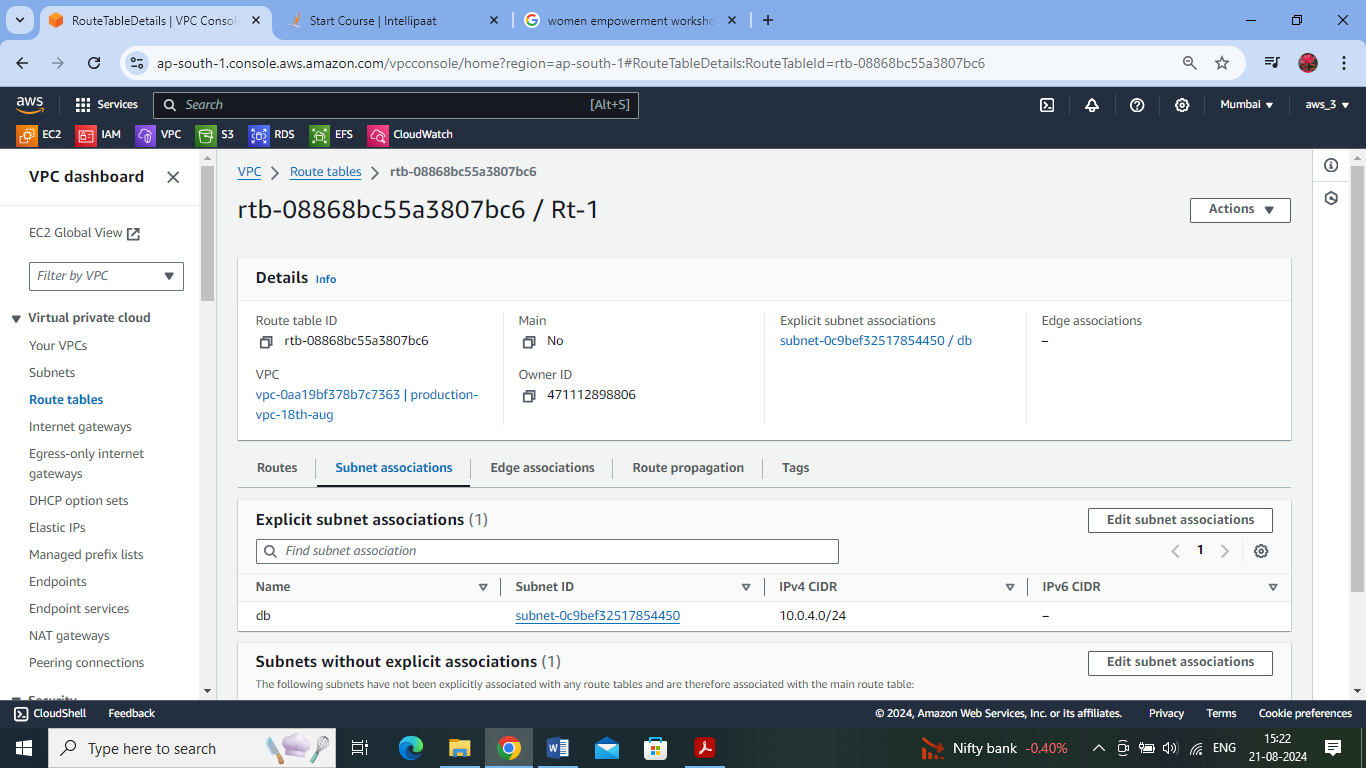
Accepted the Peering Connection:



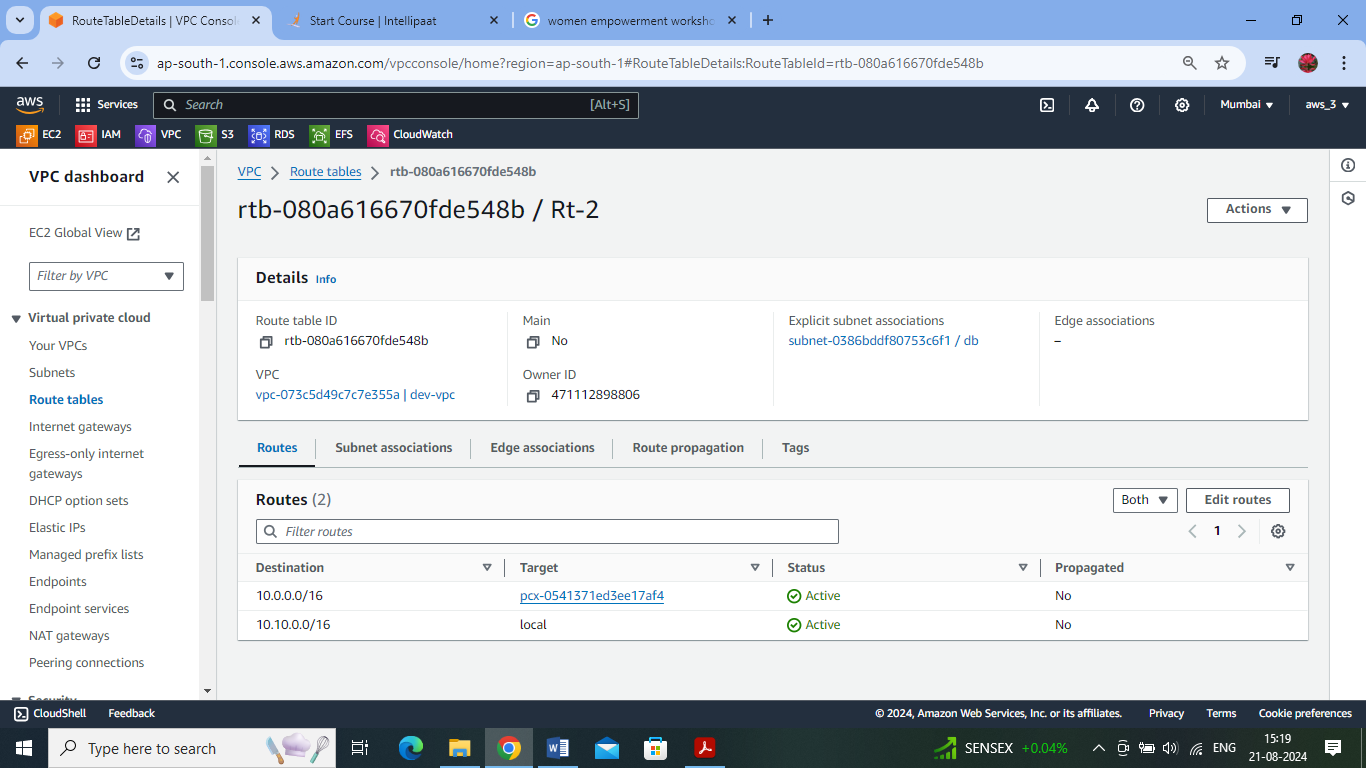
RT-1: Added the route for the peering connection – from prod vpc to dev vpc :



Associated the db subnet of prod vpc with RT-1



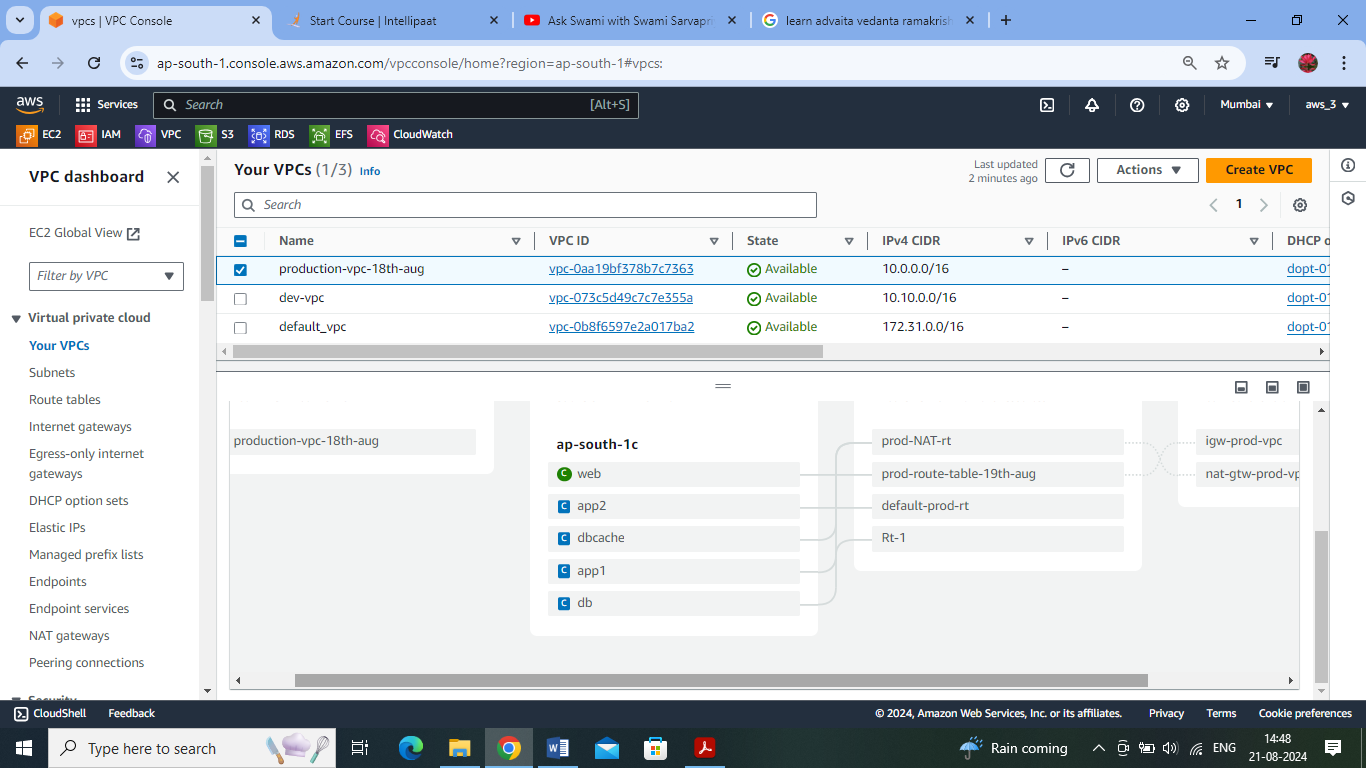
RT-2 : Added the route for the peering connection – from dev vpc to prod vpc :

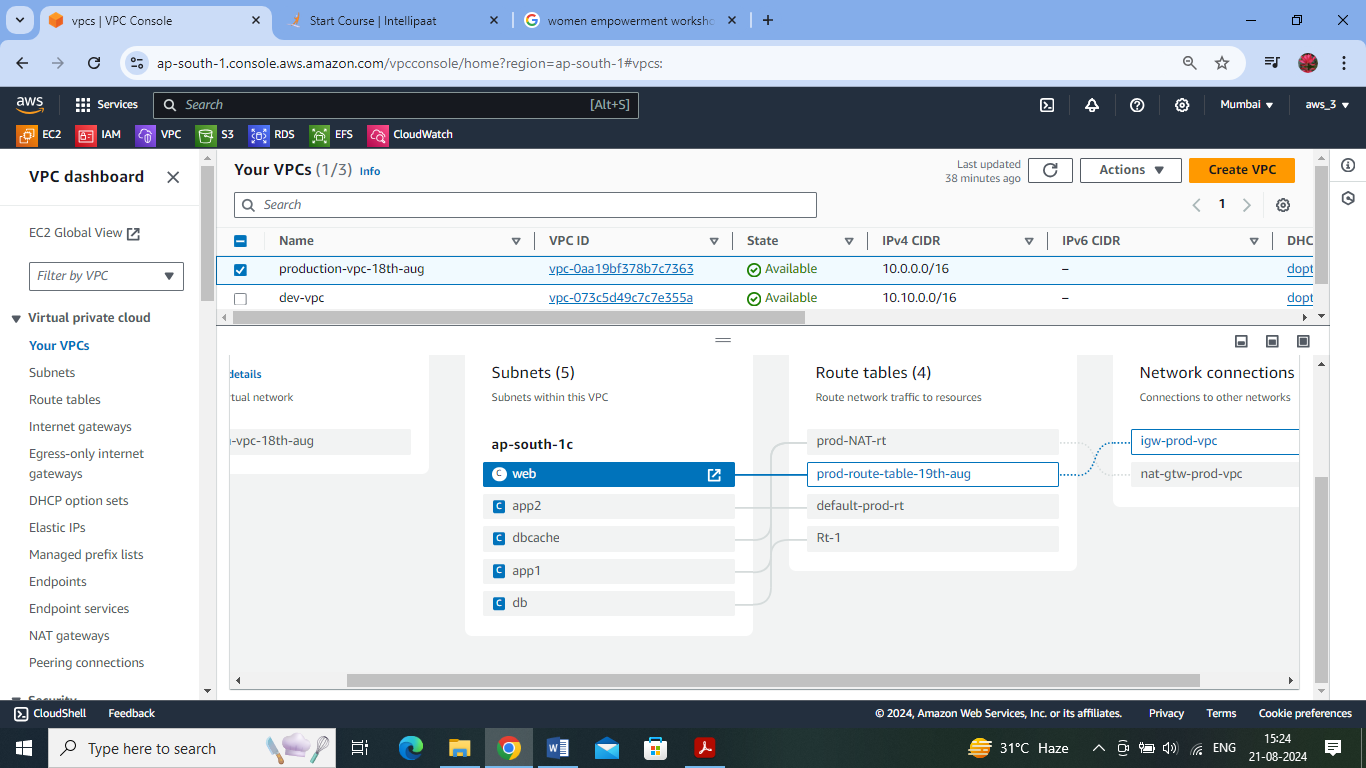


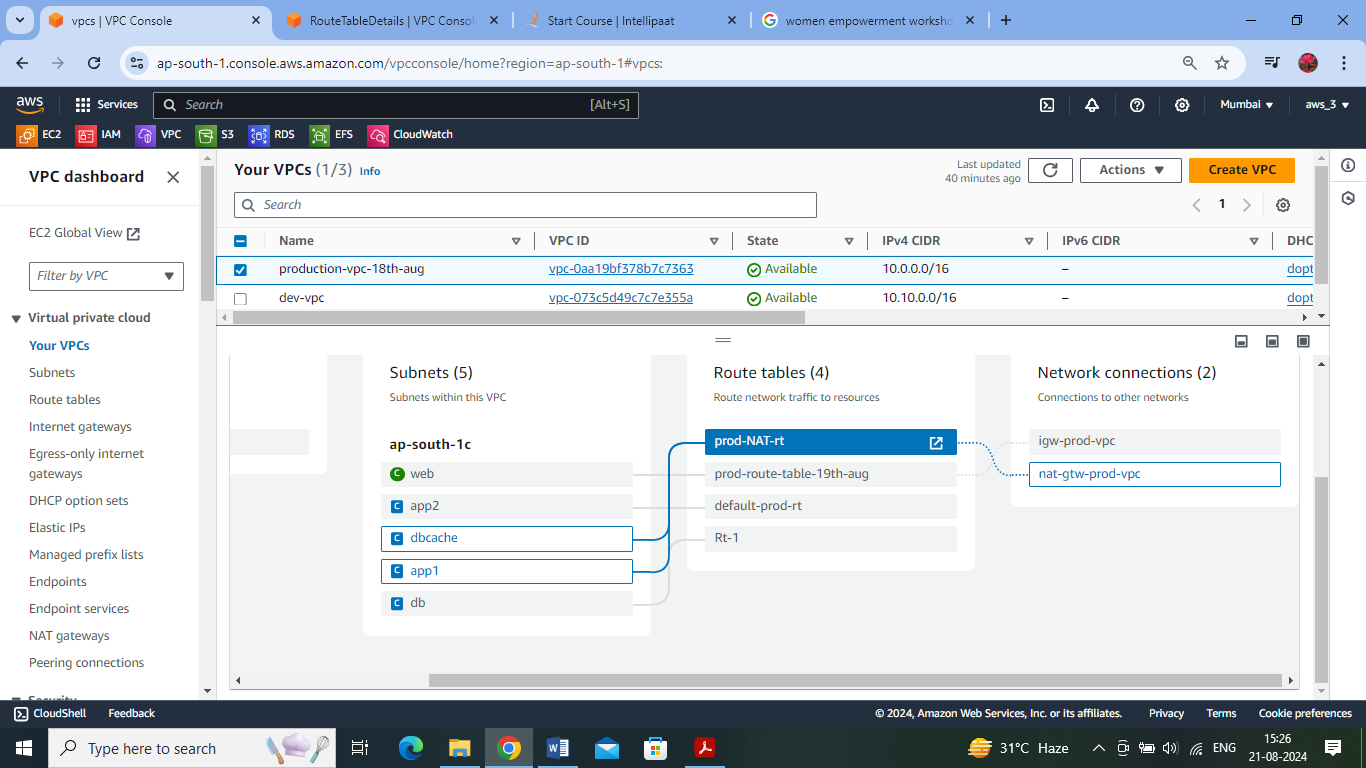
Db subnet of dev-vpc is associated with Rt-2 :

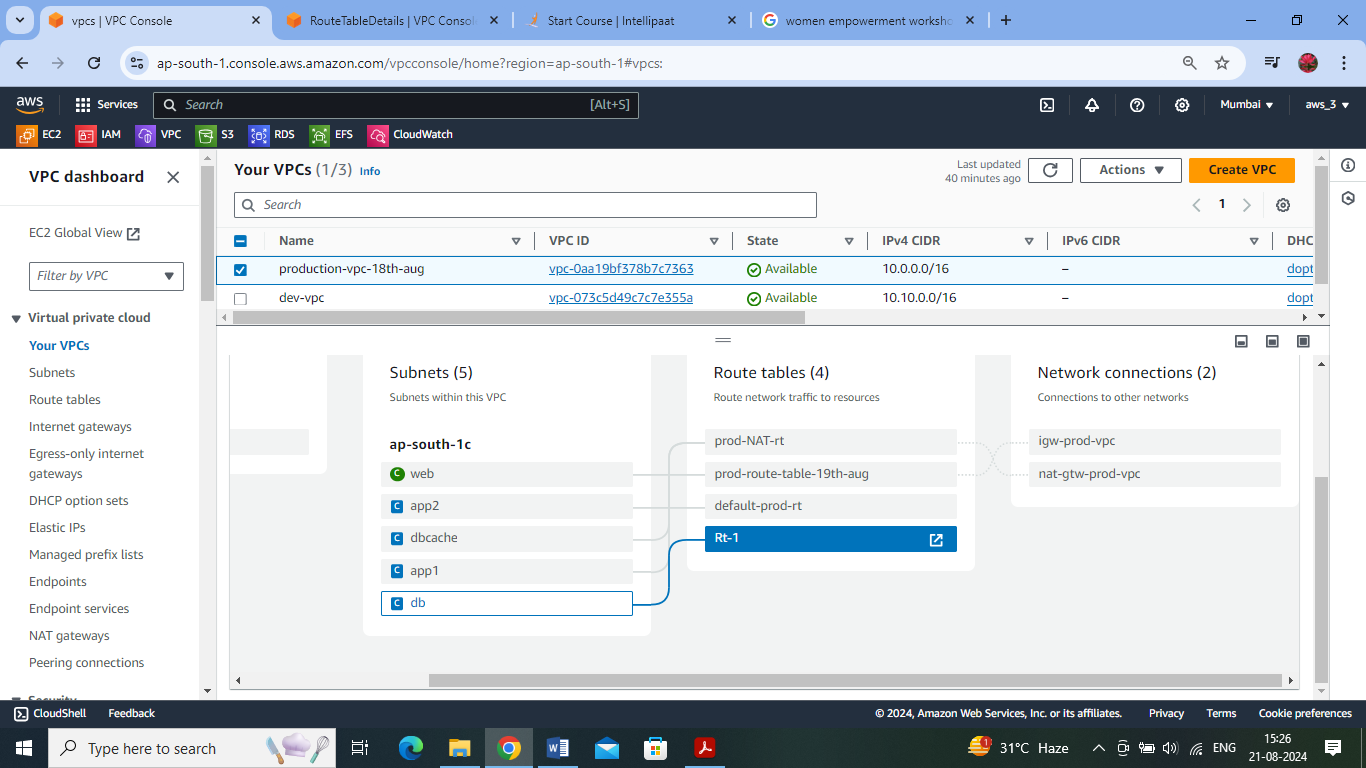


Final – Resource map of Production VPC :

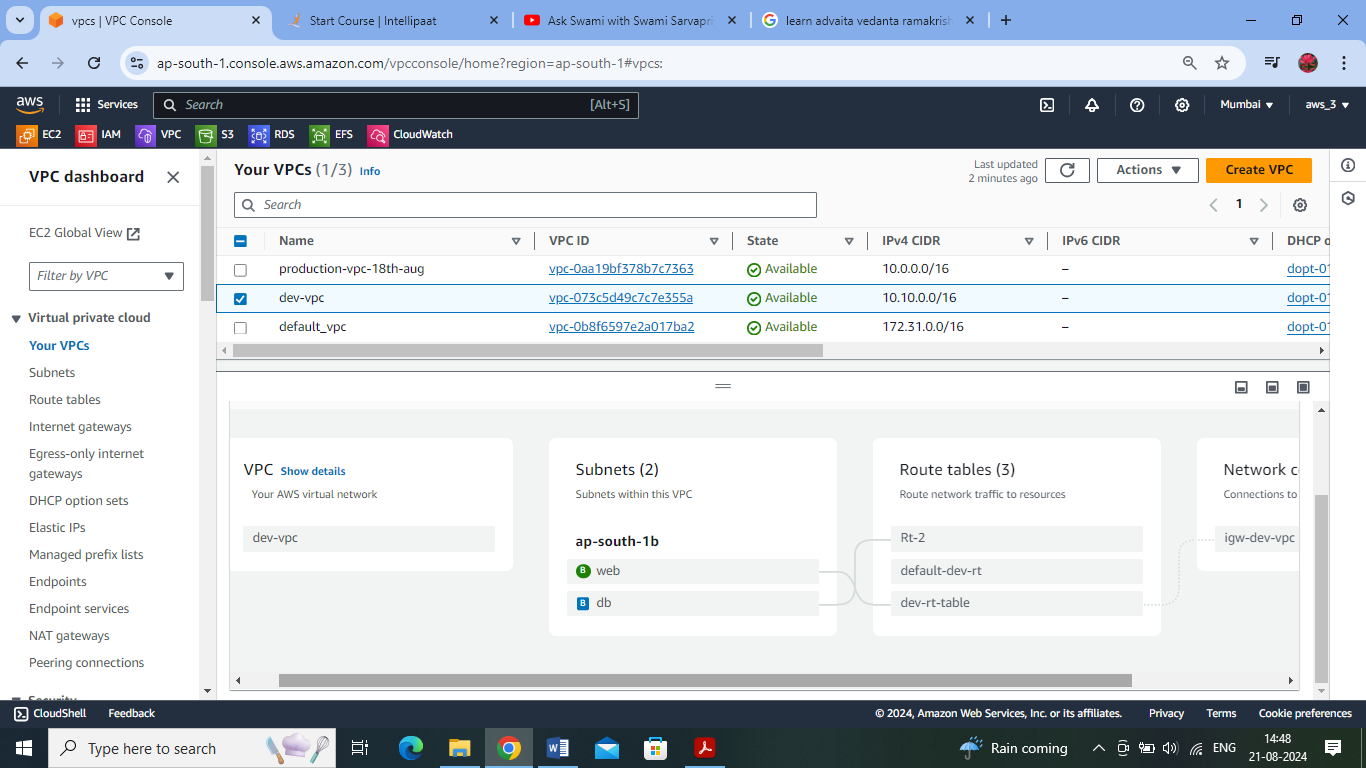


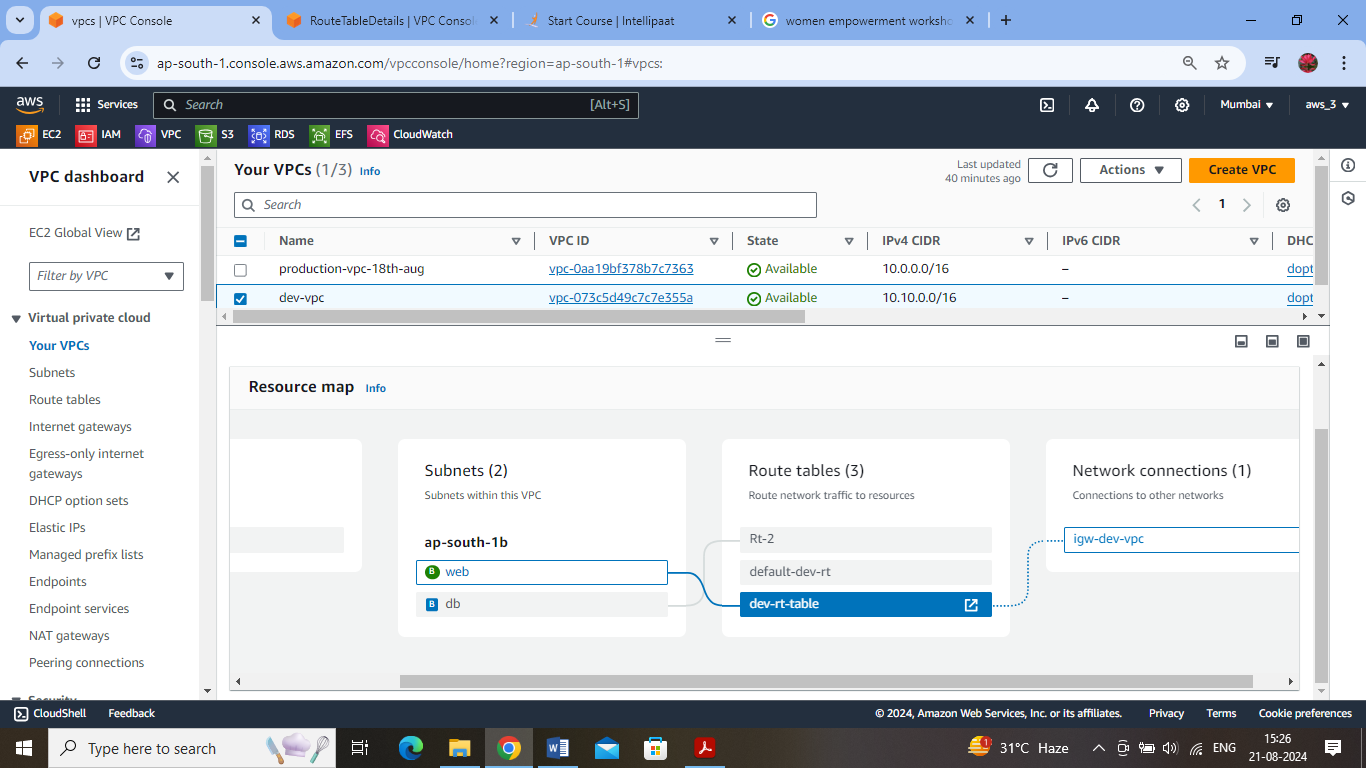


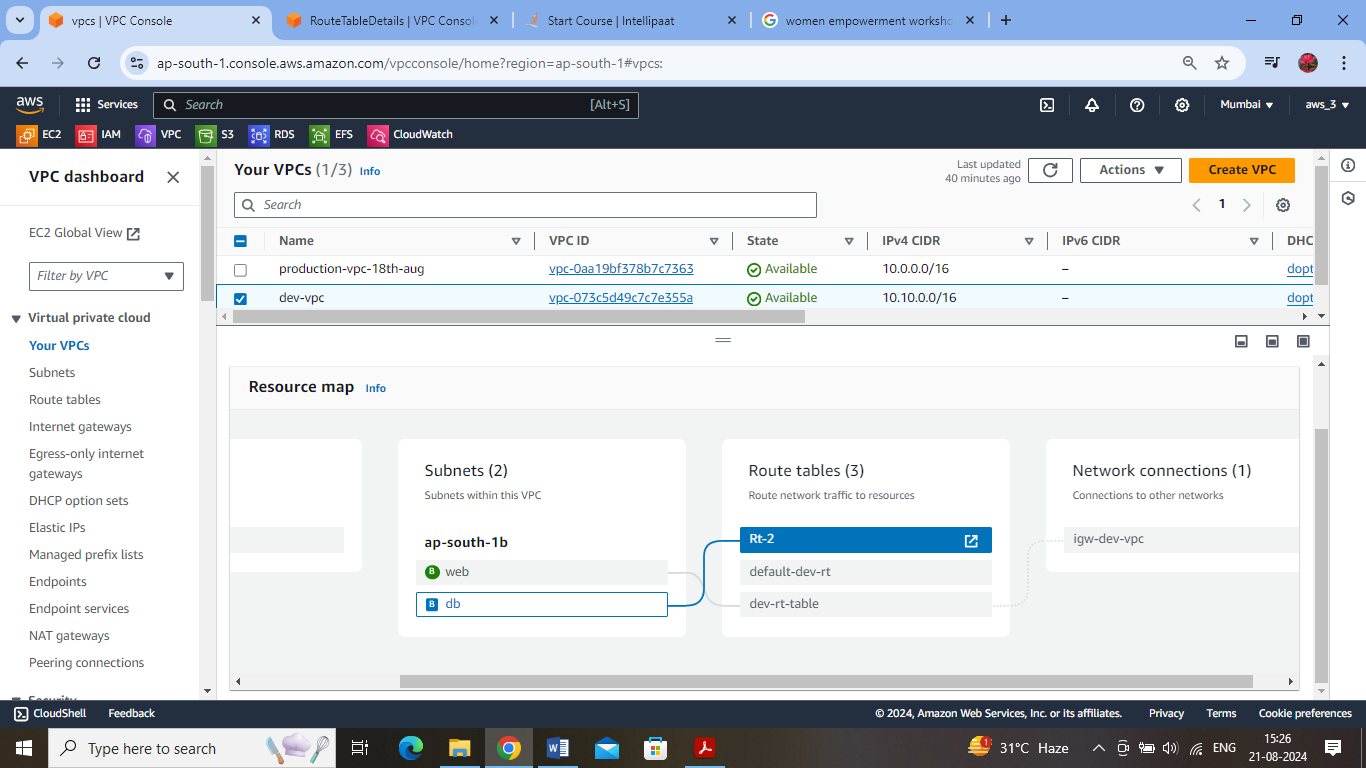




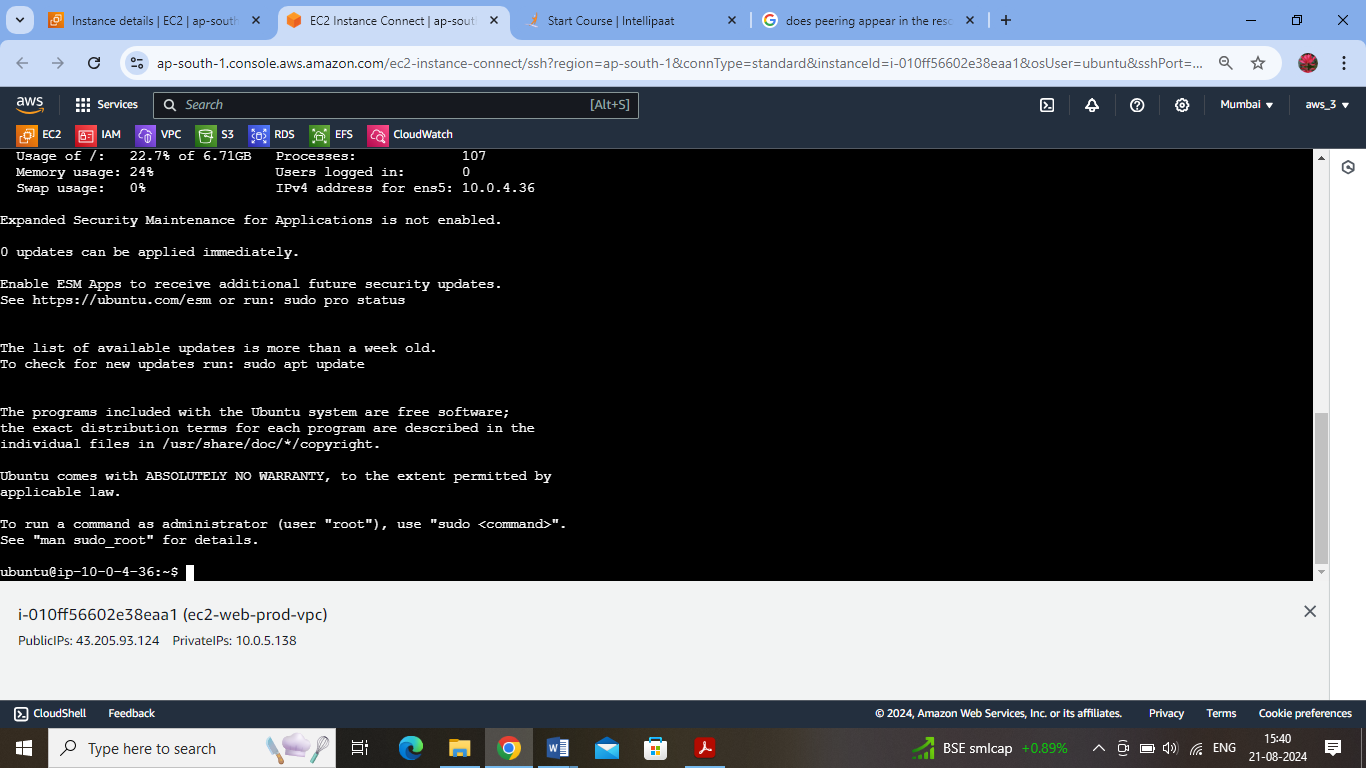
Final development vpc- resource map:



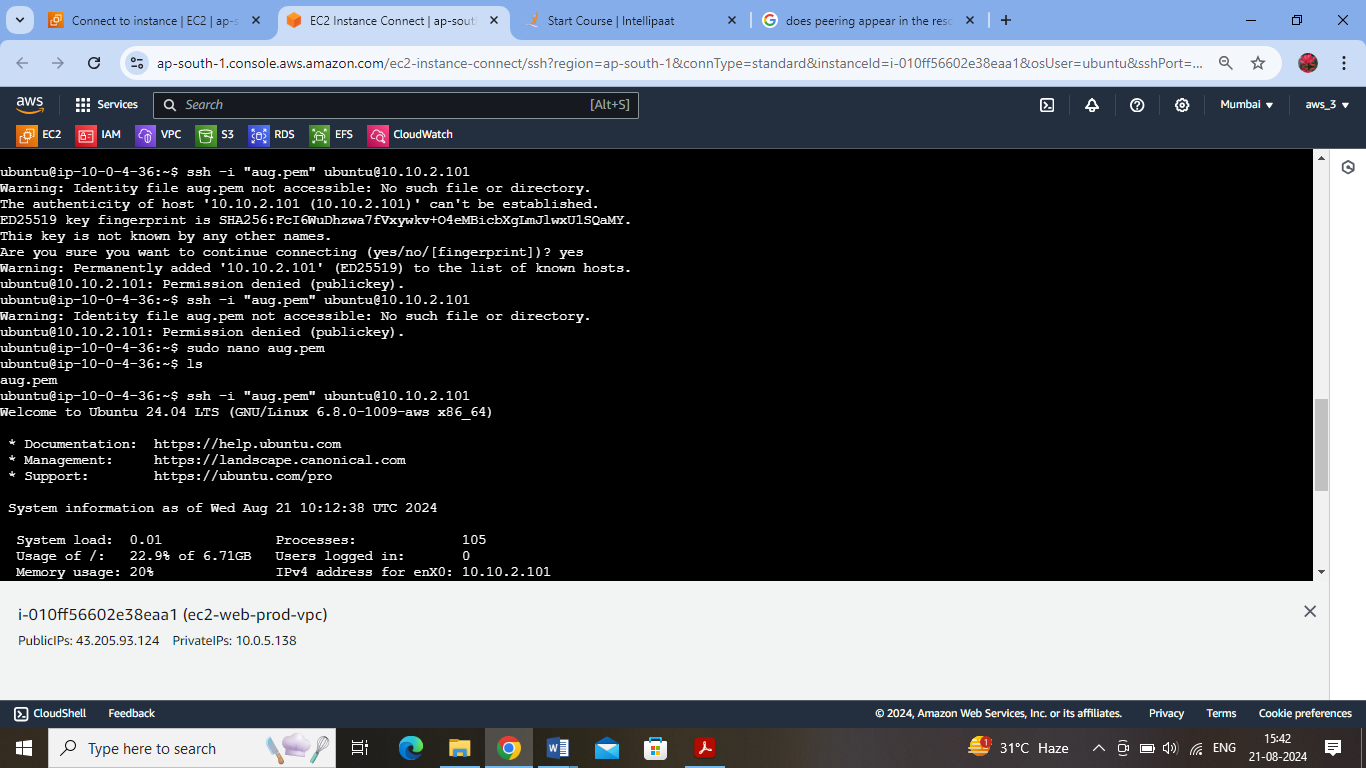


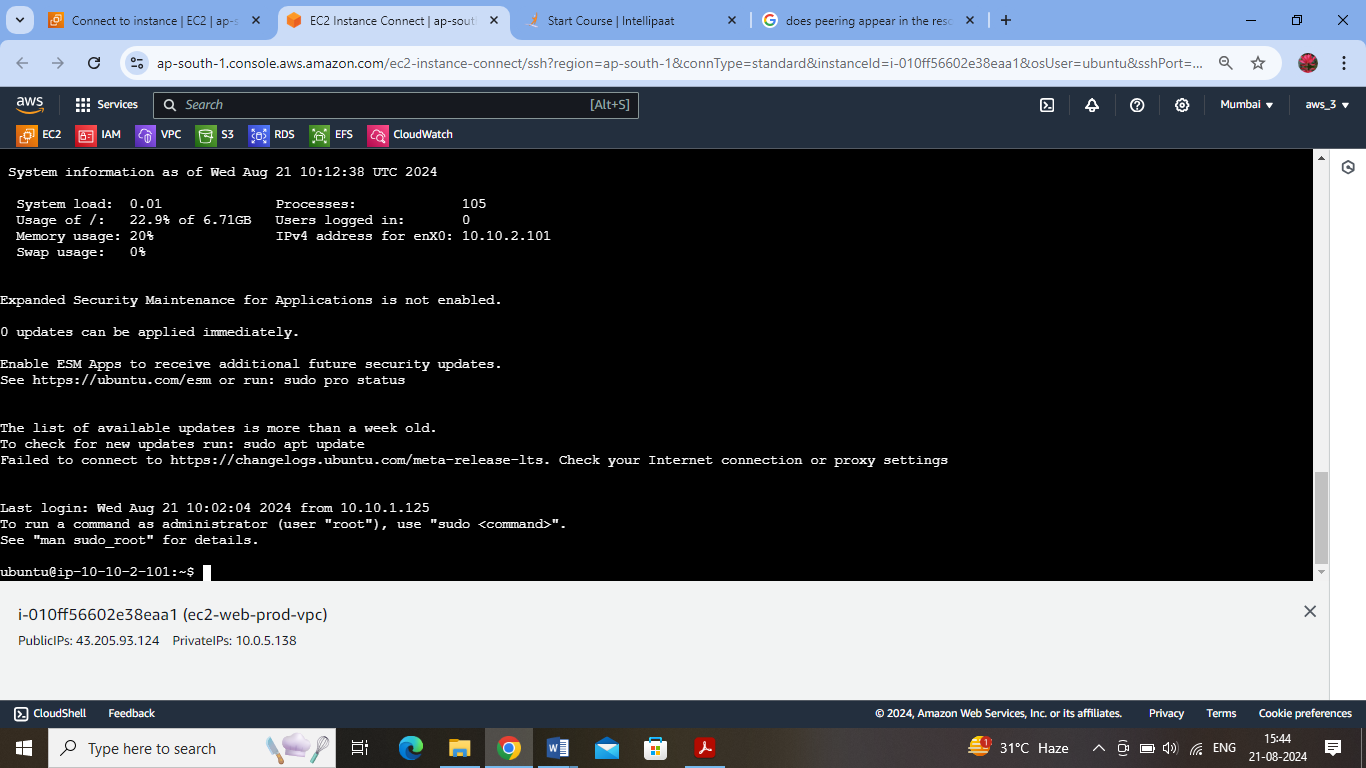


Connecting to the EC2 instance in db subnet (prod vpc) via the web subnet (prod vpc)



Connecting to the db subnet (dev-vpc) from the db subnet (prod vpc) using ssh :

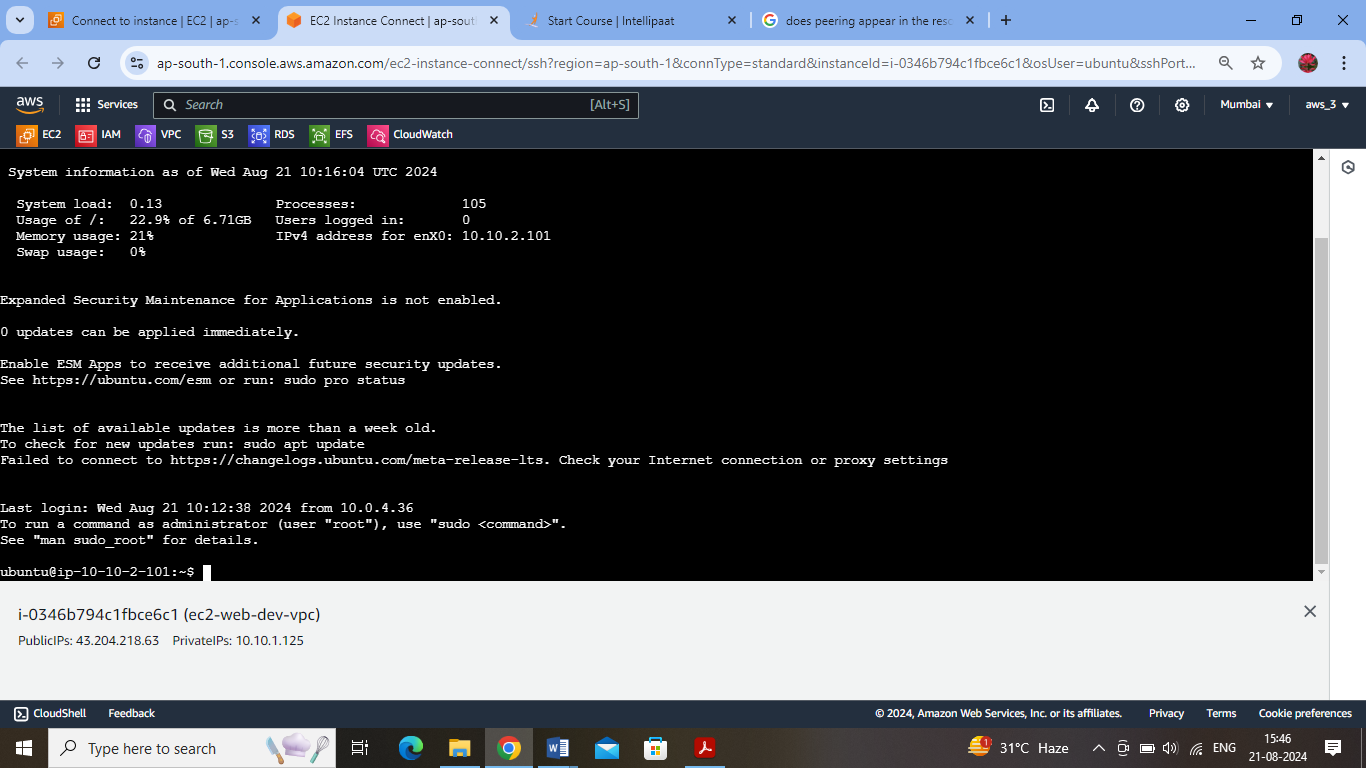




Hence the peering connection works from prod vpc -> dev vpc

Checking the vice versa :

Logged into to db subnet (dev vpc) from web subnet (dev vpc):



Logged into to db subnet (prod vpc) from db subnet of (dev vpc):

