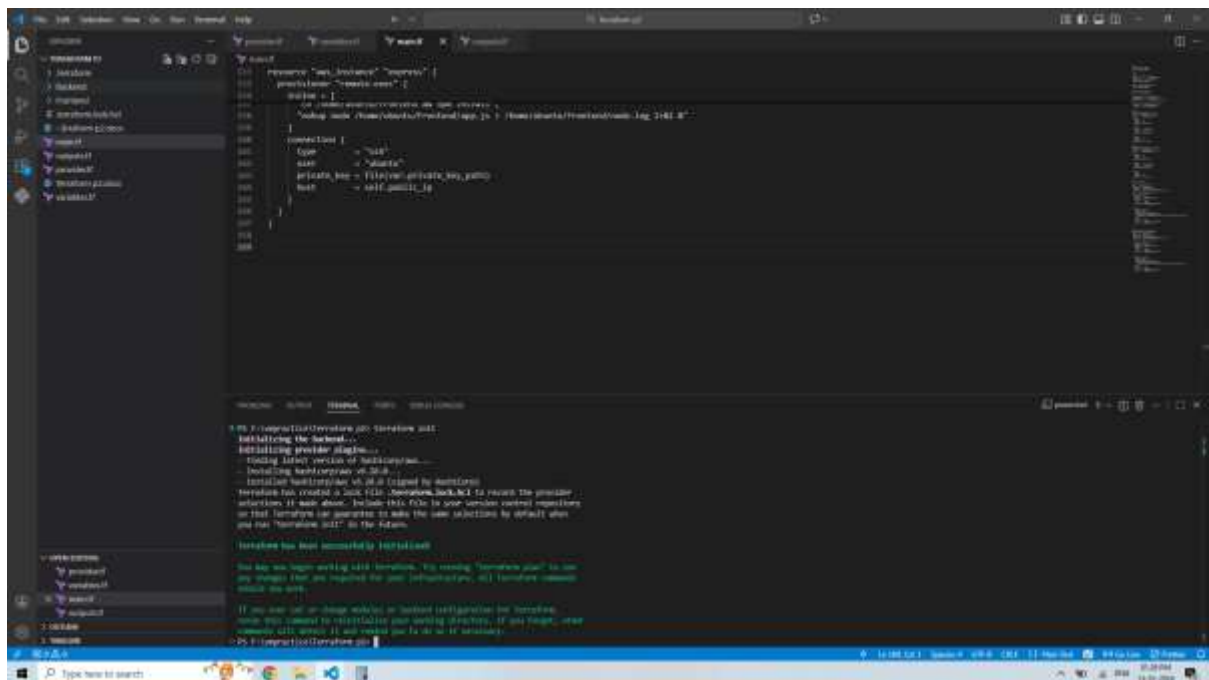
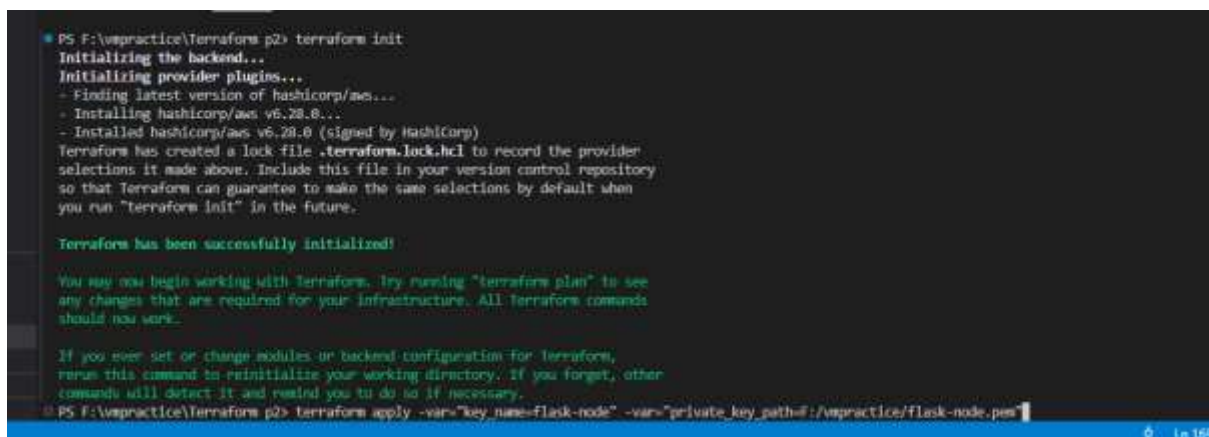


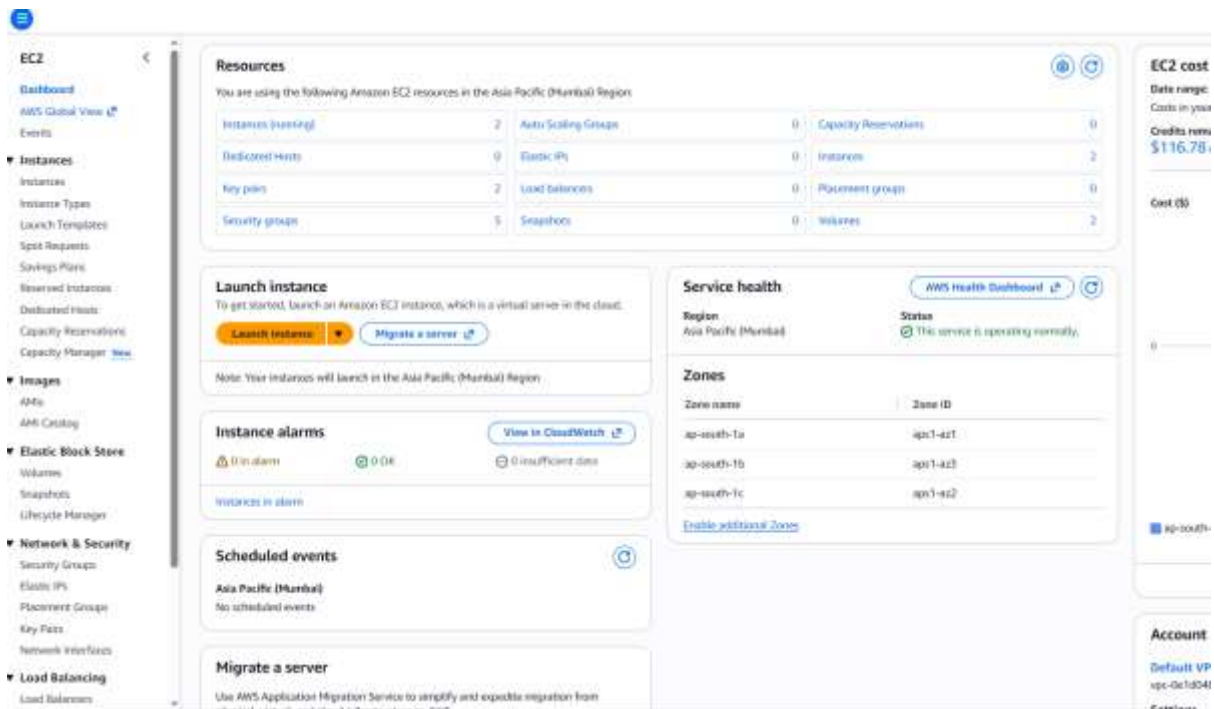
## Terraform part 2



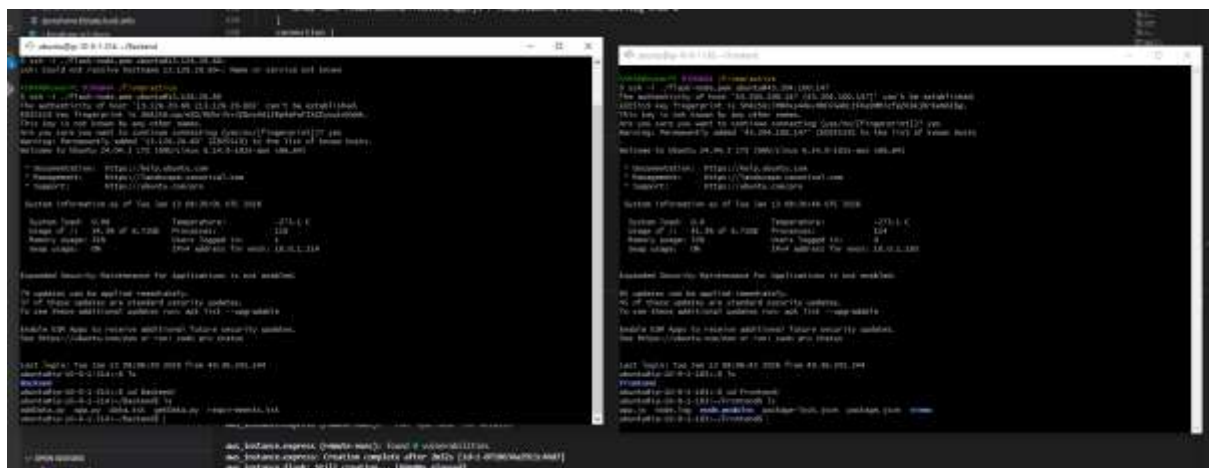
Initilize terraform working directory/folder



Apply or run terraform it create and configure EC2,subnet,VPC,Inbound rules ,outbound rules, those specified in main.tf



Two EC2 instance created and also started



Both instance configured and source code also added successfully by terraform  
we able to connect it with local using ssh

```
resource "aws_instance" "flask" {
  199   vpc_security_group_ids = [aws_security_group.flask_sg.id]
  200
  201   tags = { Name = "flask-server" }
  202
  203   provisioner "file" {
  204     source      = "backend"
  205     destination = "/home/ubuntu/backend"
  206     connection {
  207       type        = "ssh"
  208       user        = "ubuntu"
  209       private_key = file(var.private_key_path)
  210       host        = self.public_ip
  211     }
  212   }
  213
  214   provisioner "remote-exec" {
  215     inline = [
  216       "sudo apt update -y",
  217       "sudo apt install -y python3 python3-pip",
  218       "sudo apt install -y python3-flask",
  219       "python3 /home/ubuntu/backend/app.py & /home/ubuntu/backend/flask.log 20k &"
  220     ]
  221     connection {
  222       type        = "ssh"
  223       user        = "ubuntu"
  224       private_key = file(var.private_key_path)
  225       host        = self.public_ip
  226     }
  227   }
  228 }
  229
  230 aws_instance.flask (remote-exec): Service requests being deferred.
  231 aws_instance.flask (remote-exec): systemctl restart networkd-dispatcher.service
  232 aws_instance.flask (remote-exec): systemctl restart systemd-apt-daemon.service
  233
  234 aws_instance.flask (remote-exec): No containers need to be restarted.
  235
  236 aws_instance.flask (remote-exec): No user sessions are running on this
  237 aws_instance.flask (remote-exec): instance.
  238
  239 aws_instance.flask (remote-exec): No VM guests are running on this
  240 aws_instance.flask (remote-exec): hypervisor (vmx) instance on this
  241 aws_instance.flask (remote-exec): host.
  242
  243 aws_instance.flask: Creation complete after 14s [id=4-4564640000000000]
  244
  245 Apply completed successfully: 1 added, 0 changed, 1 destroyed.
  246
  247 Outputs:
  248 flask_public_ip = "40.104.104.140"
  249 flask_public_ip = "40.104.104.140"
  250 flask_public_ip = "40.104.104.140"
```

After completed create -launch – configure  
in output we get public ip of both fronted and backend

User Form

Enter your name

Enter your email

Enter your password

Submit

View Data

Also able to connect with frontend service using public ip that it return