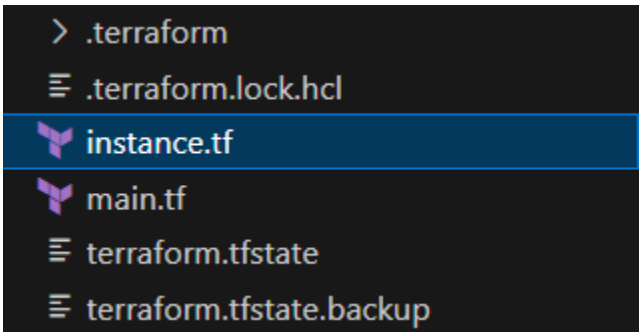
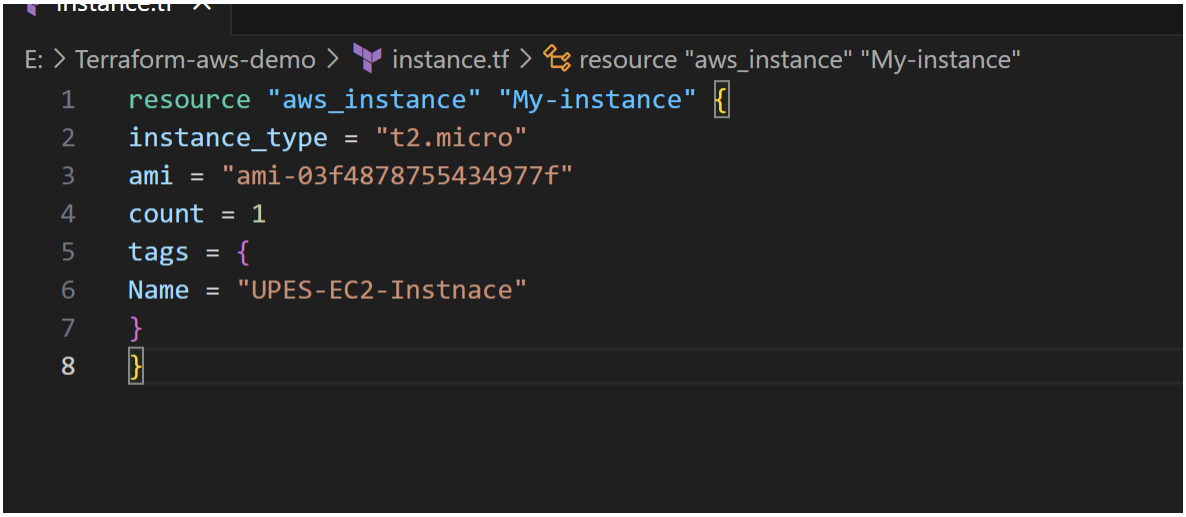
EXPERIMENT 3:

Provisioning an EC2 Instance on AWS

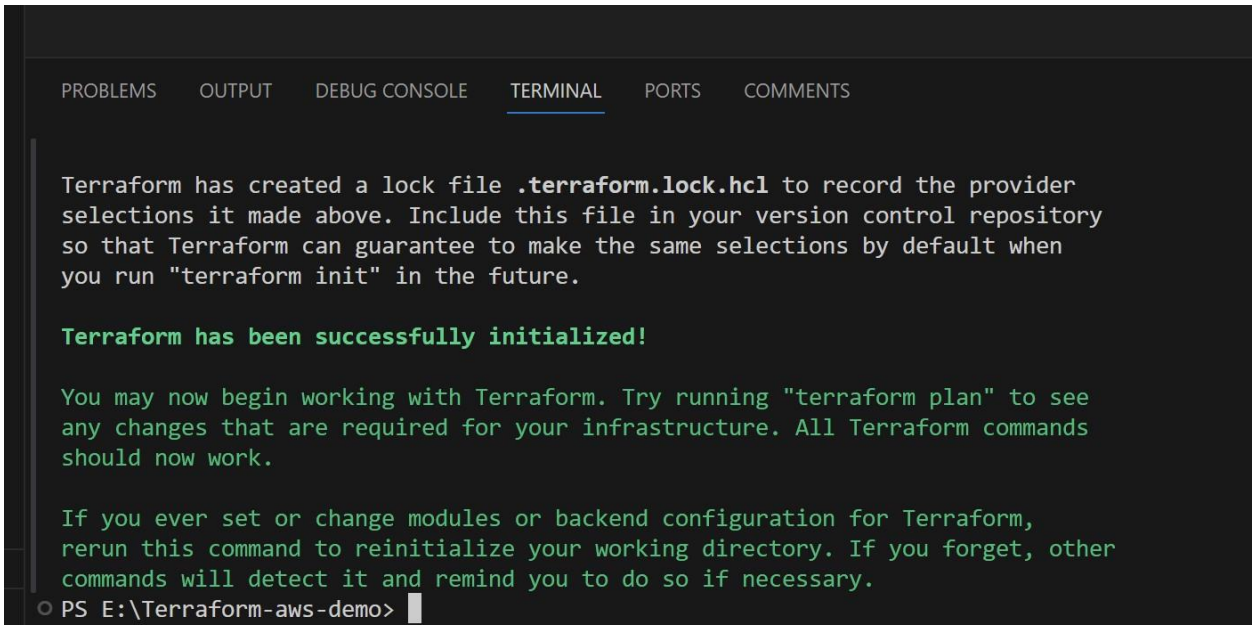
1. Create a terraform configuration file for EC2 instance called ***instance.tf***

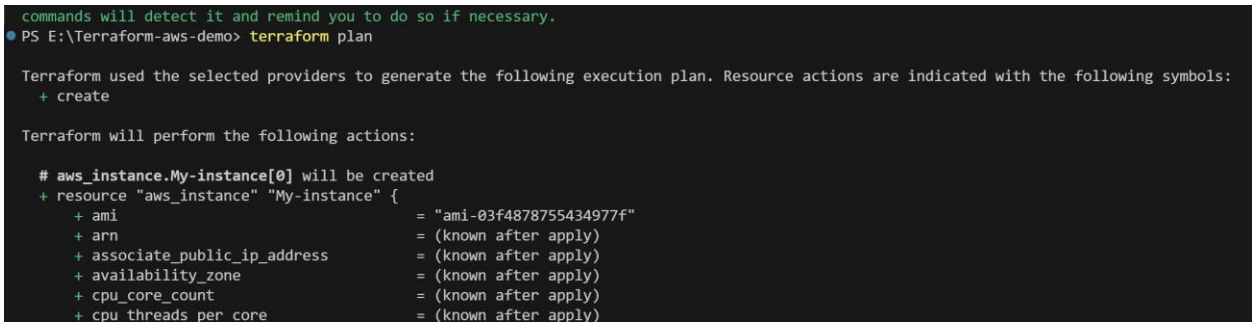


2. Note down the ami code from your AWS account.
3. Write down the following content in instance.tf

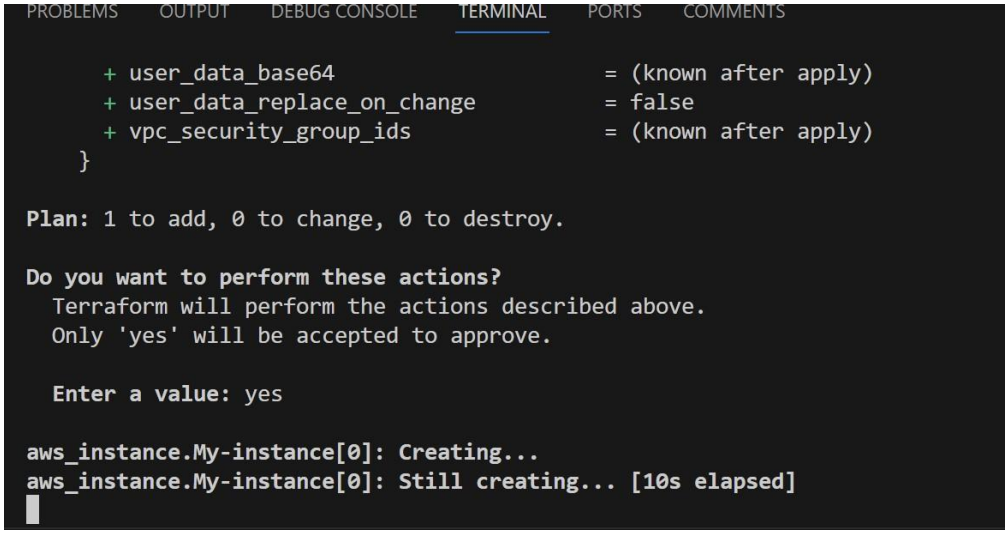


4. Review plan - Run the following command to see what terraform will do.



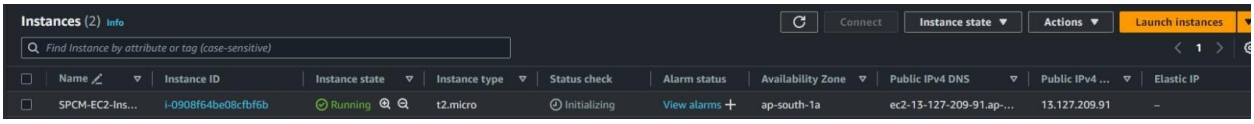


5. Apply changes - Apply the changes to create AWS resources



```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS   COMMENTS

      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)
    }

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.My-instance[0]: Creating...
aws_instance.My-instance[0]: Still creating... [10s elapsed]
```

6. Verify by logging into your AWS account and check if your resources is created.



7. When you are done with experimenting, run the following command to destroy the created resources.

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS   COMMENTS

Plan: 0 to add, 0 to change, 1 to destroy.

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.My-instance[0]: Destroying... [id=i-0063cddf43691bd0c]
aws_instance.My-instance[0]: Still destroying... [id=i-0063cddf43691bd0c, 10s elapsed]
aws_instance.My-instance[0]: Still destroying... [id=i-0063cddf43691bd0c, 20s elapsed]
aws_instance.My-instance[0]: Still destroying... [id=i-0063cddf43691bd0c, 30s elapsed]
aws_instance.My-instance[0]: Destruction complete after 31s

Destroy complete! Resources: 1 destroyed.
PS E:\Terraform-aws-demo>
```