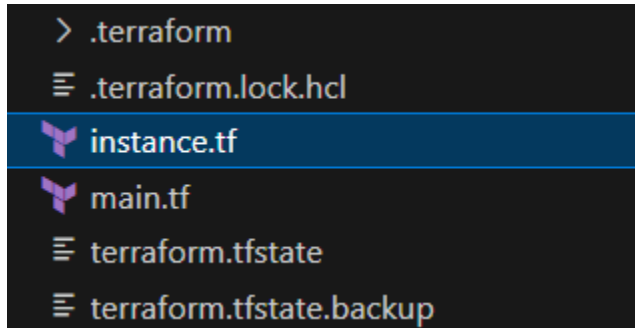


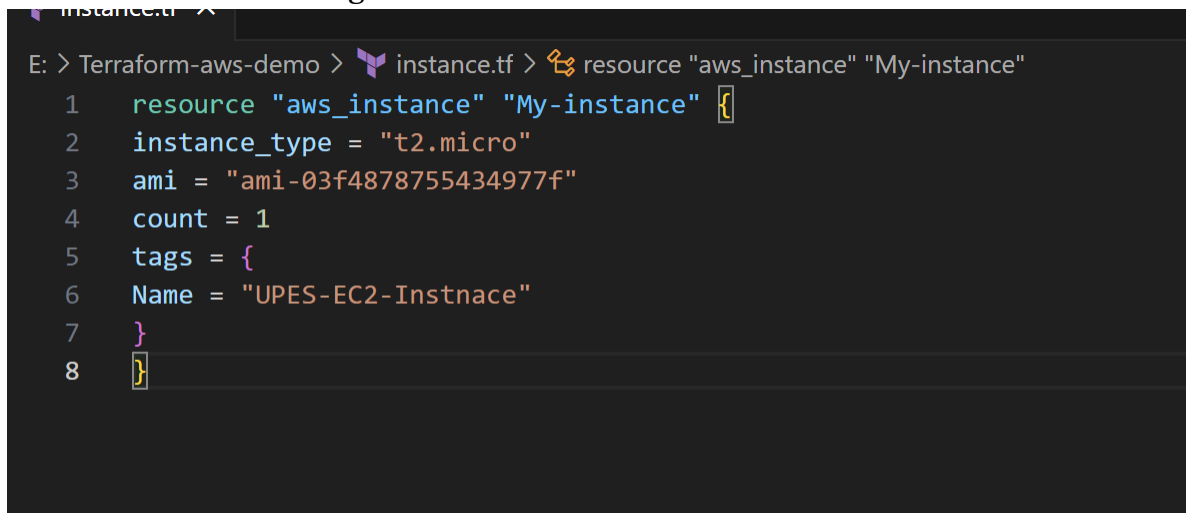
## EXPERIMENT 3:

### 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.

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

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
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.
PS E:\Terraform-aws-demo>
```

```
commands will detect it and remind you to do so if necessary.
PS E:\Terraform-aws-demo> 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.My-instance[0] will be created
+ resource "aws_instance" "My-instance" {
  + ami                  = "ami-03f4878755434977f"
  + arn                  = (known after apply)
  + associate_public_ip_address = (known after apply)
  + availability_zone      = (known after apply)
  + cpu_core_count         = (known after apply)
  + cpu_threads_per_core   = (known after apply)
}
```

## 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]
```

```
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.

Instances (2) <small>Info</small>									
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>									
<input type="checkbox"/>	Name <small>↗</small>	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
<input type="checkbox"/>	SPCM-EC2-Ins...	i-0908f64be08cfb6b	<span>Running</span>	t2.micro	Initializing	<a href="#">View alarms</a>	ap-south-1a	ec2-13-127-209-91.ap-...	13.127.209.91

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>
```