**Terraform Output Values**

In the output attribute, insert attribute name, description and value=”resource type.local name of the resource. Type (id, ip etc)

Output values in the cli

# Define Output Values

# Attribute Reference: EC2 Instance Public IP

output "ec2\_instance\_publicip" {

  description = "EC2 Instance Public IP"

  value = aws\_instance.my-ec2-vm.public\_ip

}

# Argument Reference: EC2 Instance Private IP

output "ec2\_instance\_privateip" {

  description = "EC2 Instance Private IP"

  value = aws\_instance.my-ec2-vm.private\_ip

}

# Argument Reference: Security Groups associated to EC2 Instance

output "ec2\_security\_groups" {

  description = "List Security Groups associated with EC2 Instance"

  value = aws\_instance.my-ec2-vm.security\_groups

}

# Attribute Reference - Create Public DNS URL with http:// appended

output "ec2\_publicdns" {

  description = "Public DNS URL of an EC2 Instance"

  value = "http://${aws\_instance.my-ec2-vm.public\_dns}"

  #sensitive = true   #Uncomment it during step-04 execution

}

Terraform apply

A screenshot of a computer

Description automatically generated

You can also write

terraform output

for the same output

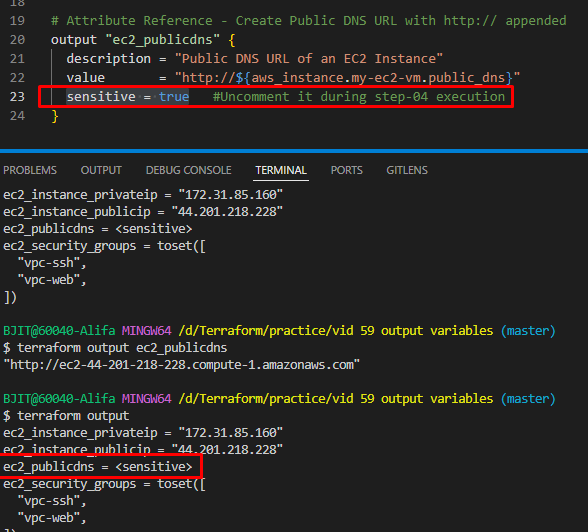
terraform output ec2\_publicdns

to see ec2 public dns

**Suppressing Sensitive Values in Output**

sensitive = true

lets say the ec2 public dns is sensitive, so in the outputs file in the attribute of the dns, we make sensitive is true.



But in terraformtfstate file you will be able to see the sensitive data, so when terraform

# Query using terraform output

terraform output ec2\_publicdns

Observation: You should get non-redacted original value from terraform.tfstate file