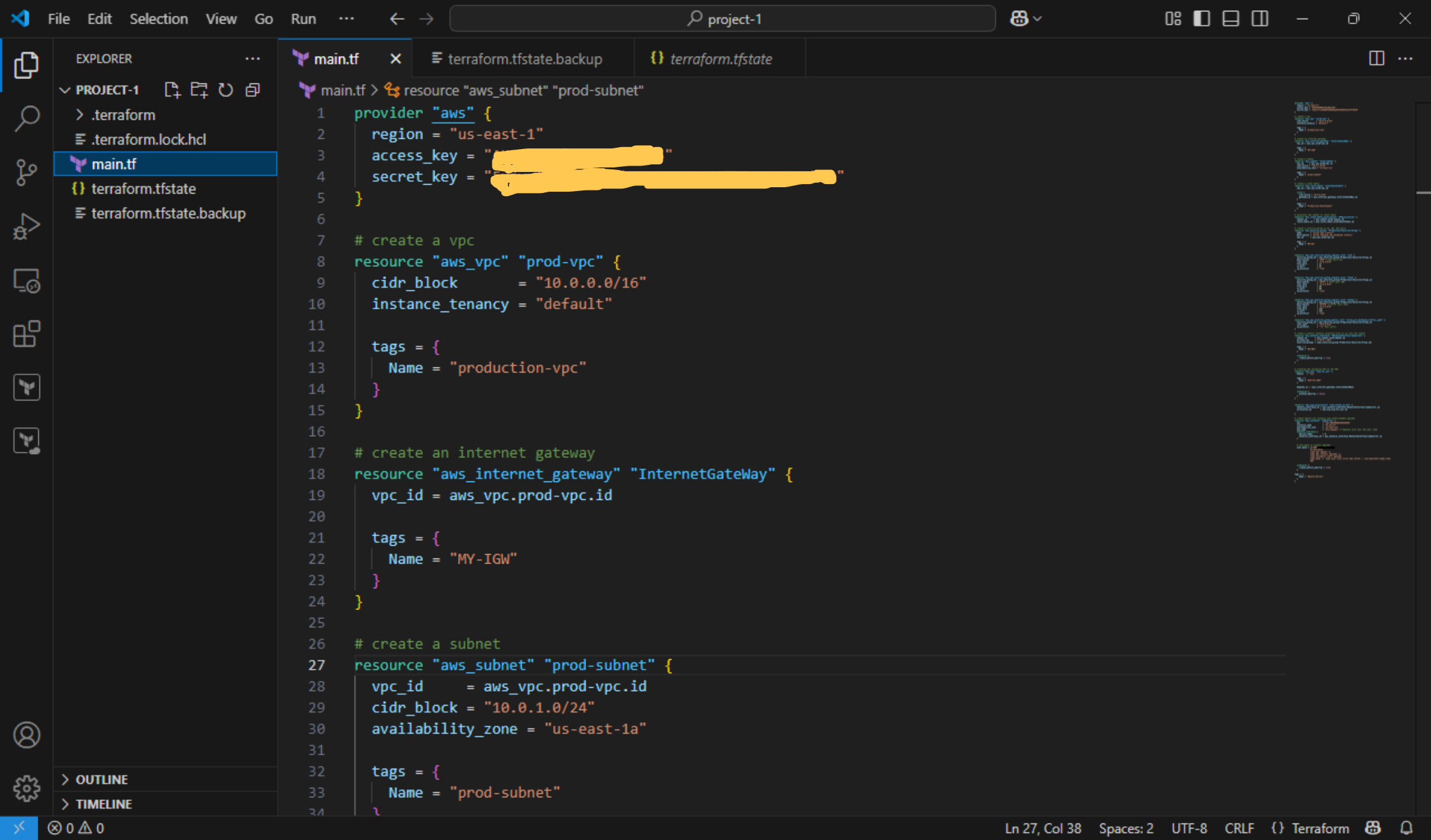
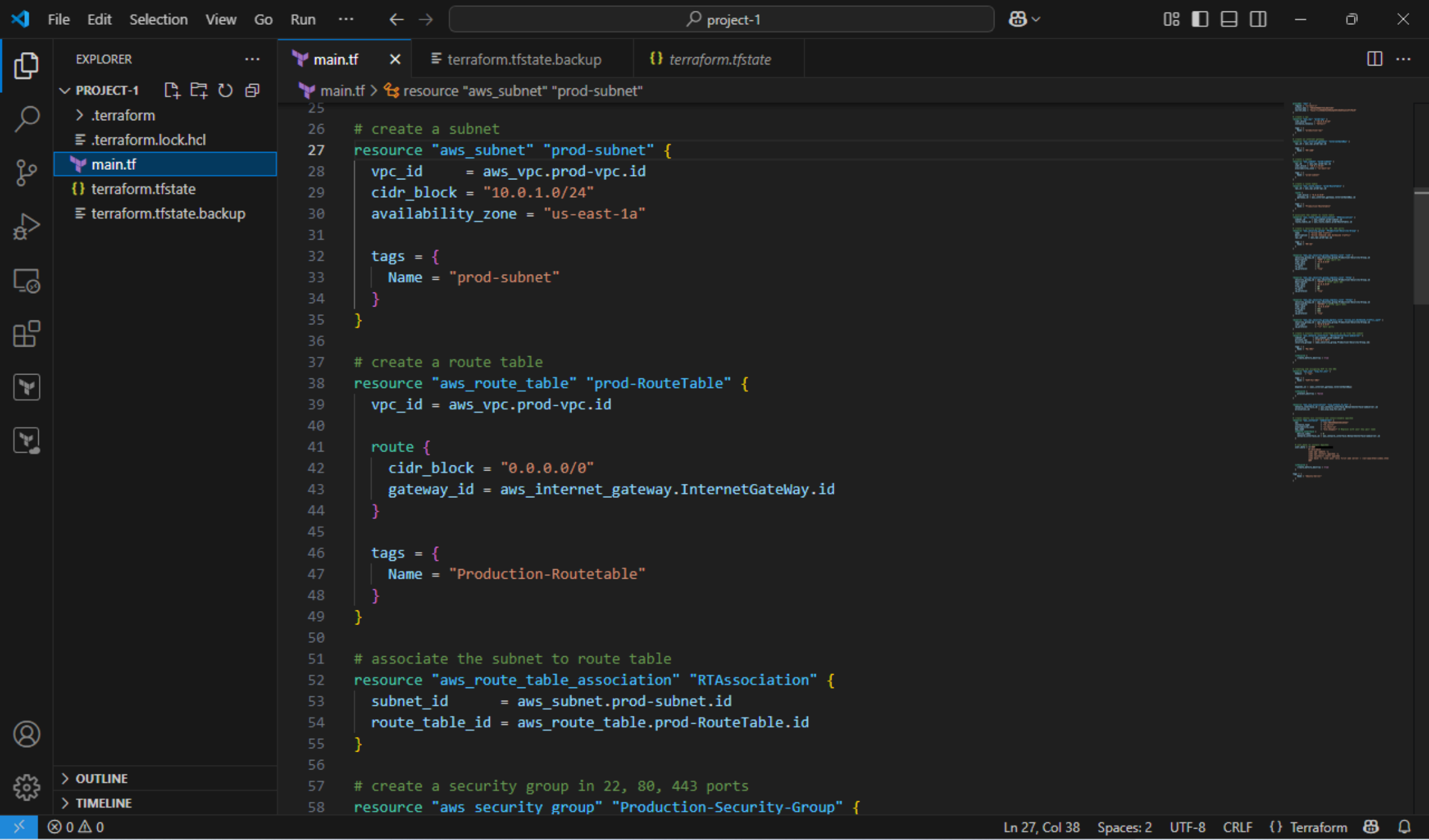
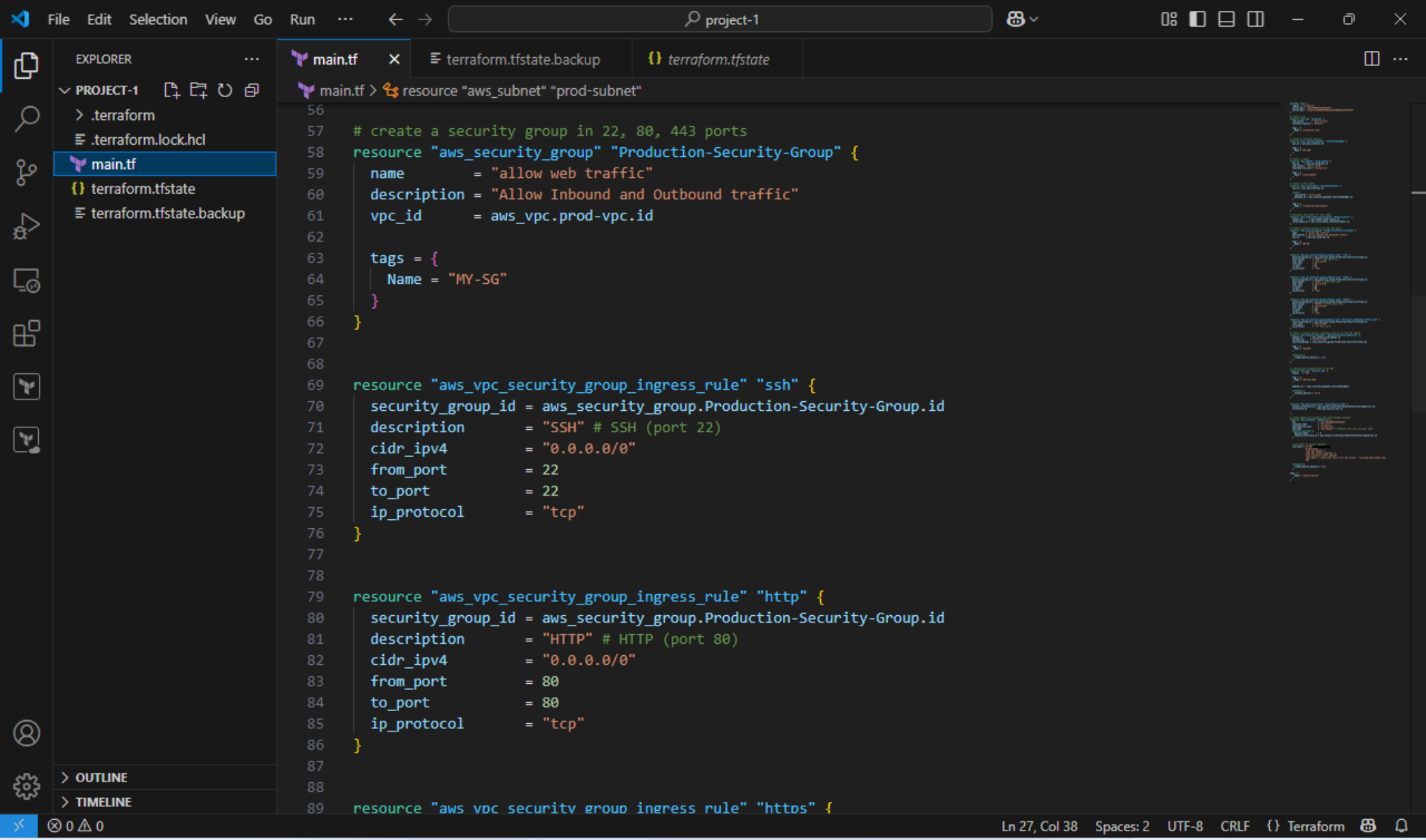
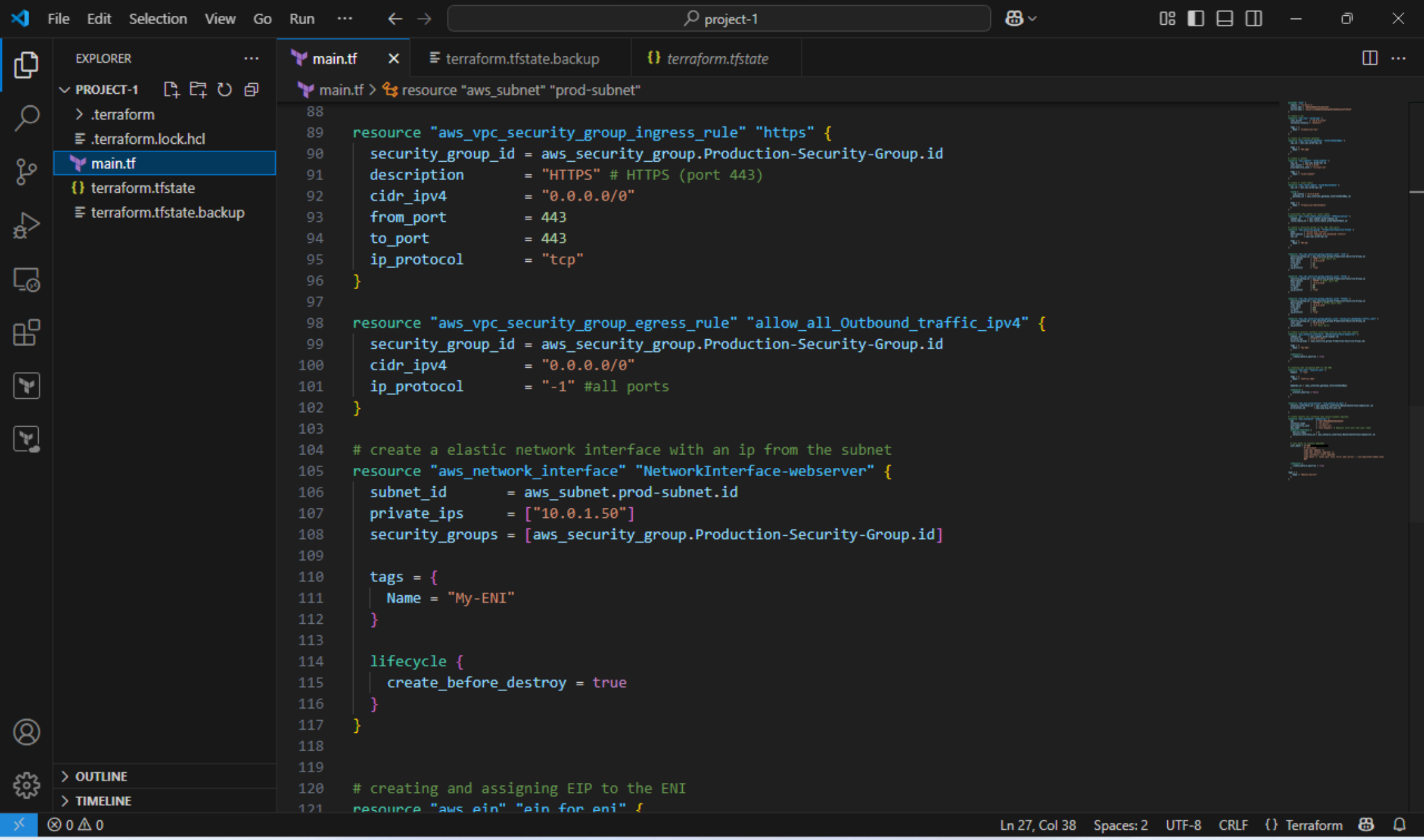
HashiCorp

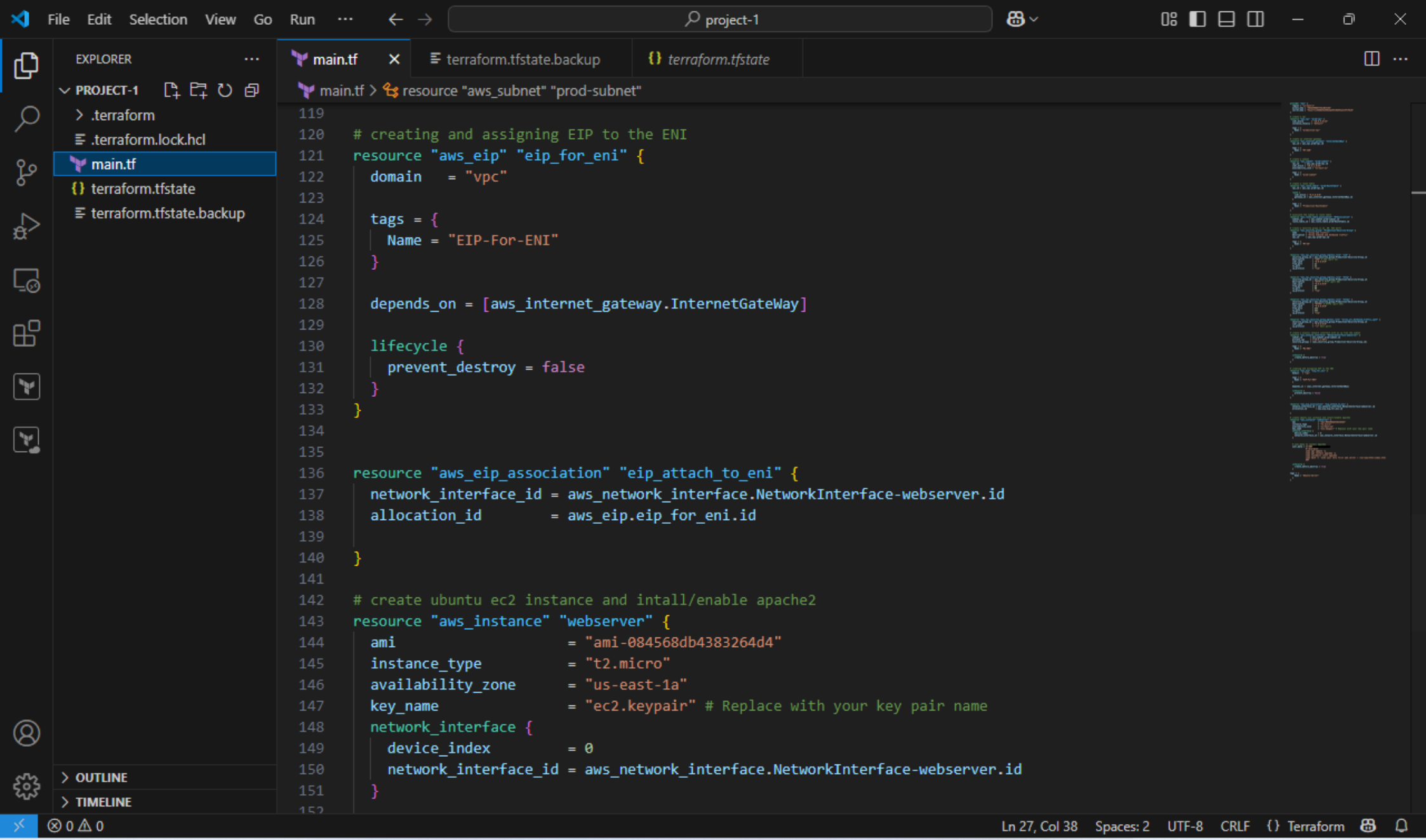
Terraform

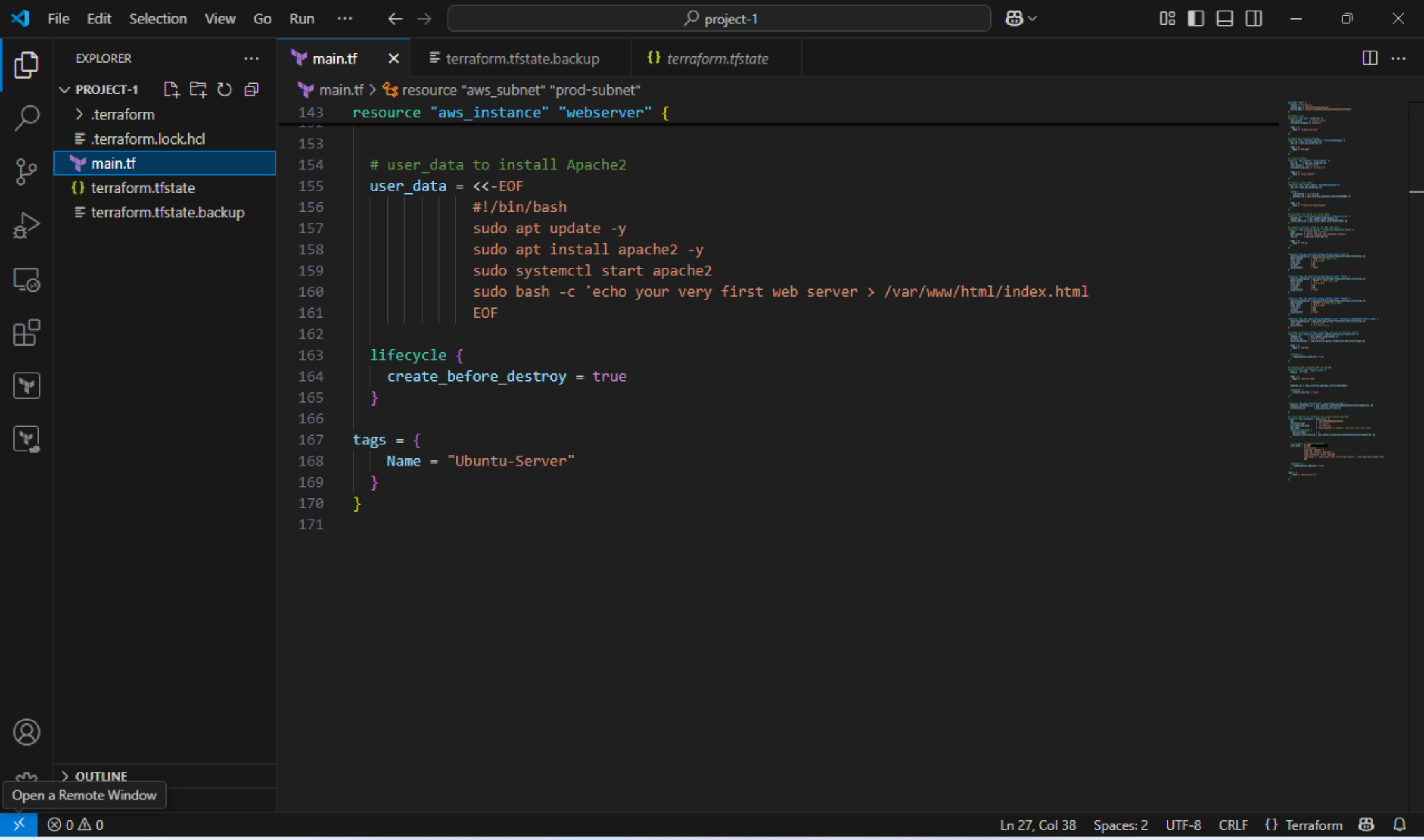

File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
main.tf
terraform.tfstate.backup
terraform.tfstate
main.tf > resource "aws_subnet" "prod-subnet"
provider "aws" {
region = "us-east-1"
access_key = "[redacted]"
secret_key = "[redacted]"
}
# create a vpc
resource "aws_vpc" "prod-vpc" {
cidr_block = "10.0.0.0/16"
instance_tenancy = "default"
tags = {
Name = "production-vpc"
}
}
# create an internet gateway
resource "aws_internet_gateway" "InternetGateWay" {
vpc_id = aws_vpc.prod-vpc.id
tags = {
Name = "MY-IGW"
}
}
# create a subnet
resource "aws_subnet" "prod-subnet" {
vpc_id = aws_vpc.prod-vpc.id
cidr_block = "10.0.1.0/24"
availability_zone = "us-east-1a"
tags = {
Name = "prod-subnet"
OUTLINE
TIMELINE
Ln 27, Col 38
Spaces: 2
UTF-8
CRLF
Terraform


File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
main.tf
terraform.tfstate.backup
terraform.tfstate
main.tf > resource "aws_subnet" "prod-subnet"
25
26 # create a subnet
27 resource "aws_subnet" "prod-subnet" {
28 vpc_id = aws_vpc.prod-vpc.id
29 cidr_block = "10.0.1.0/24"
30 availability_zone = "us-east-1a"
31
32 tags = {
33 Name = "prod-subnet"
34 }
35 }
36
37 # create a route table
38 resource "aws_route_table" "prod-RouteTable" {
39 vpc_id = aws_vpc.prod-vpc.id
40
41 route {
42 cidr_block = "0.0.0.0/0"
43 gateway_id = aws_internet_gateway.InternetGateWay.id
44 }
45
46 tags = {
47 Name = "Production-Routetable"
48 }
49 }
50
51 # associate the subnet to route table
52 resource "aws_route_table_association" "RTAssociation" {
53 subnet_id = aws_subnet.prod-subnet.id
54 route_table_id = aws_route_table.prod-RouteTable.id
55 }
56
57 # create a security group in 22, 80, 443 ports
58 resource "aws security group" "Production-Security-Group" {
OUTLINE
TIMELINE
0 0
Ln 27, Col 38
Spaces: 2
UTF-8
CRLF
Terraform


File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
main.tf
terraform.tfstate.backup
terraform.tfstate
main.tf > resource "aws_subnet" "prod-subnet"
# create a security group in 22, 80, 443 ports
resource "aws_security_group" "Production-Security-Group" {
name = "allow web traffic"
description = "Allow Inbound and Outbound traffic"
vpc_id = aws_vpc.prod-vpc.id
tags = {
Name = "MY-SG"
}
}
resource "aws_vpc_security_group_ingress_rule" "ssh" {
security_group_id = aws_security_group.Production-Security-Group.id
description = "SSH" # SSH (port 22)
cidr_ipv4 = "0.0.0.0/0"
from_port = 22
to_port = 22
ip_protocol = "tcp"
}
resource "aws_vpc_security_group_ingress_rule" "http" {
security_group_id = aws_security_group.Production-Security-Group.id
description = "HTTP" # HTTP (port 80)
cidr_ipv4 = "0.0.0.0/0"
from_port = 80
to_port = 80
ip_protocol = "tcp"
}
resource "aws_vpc_security_group_ingress_rule" "https" {
OUTLINE
TIMELINE
Ln 27, Col 38
Spaces: 2
UTF-8
CRLF
Terraform


File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
main.tf
terraform.tfstate.backup
terraform.tfstate
main.tf > resource "aws_subnet" "prod-subnet"
88
89 resource "aws_vpc_security_group_ingress_rule" "https" {
90 security_group_id = aws_security_group.Production-Security-Group.id
91 description = "HTTPS" # HTTPS (port 443)
92 cidr_ipv4 = "0.0.0.0/0"
93 from_port = 443
94 to_port = 443
95 ip_protocol = "tcp"
96 }
97
98 resource "aws_vpc_security_group_egress_rule" "allow_all_Outbound_traffic_ipv4" {
99 security_group_id = aws_security_group.Production-Security-Group.id
100 cidr_ipv4 = "0.0.0.0/0"
101 ip_protocol = "-1" #all ports
102 }
103
104 # create a elastic network interface with an ip from the subnet
105 resource "aws_network_interface" "NetworkInterface-webserver" {
106 subnet_id = aws_subnet.prod-subnet.id
107 private_ips = ["10.0.1.50"]
108 security_groups = [aws_security_group.Production-Security-Group.id]
109
110 tags = {
111 Name = "My-ENI"
112 }
113
114 lifecycle {
115 create_before_destroy = true
116 }
117 }
118
119
120 # creating and assigning EIP to the ENI
121 resource "aws_eip" "eip_for_eni" {
OUTLINE
TIMELINE
0 0
Ln 27, Col 38
Spaces: 2
UTF-8
CRLF
Terraform


File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
main.tf
terraform.tfstate.backup
terraform.tfstate
main.tf > resource "aws_subnet" "prod-subnet"
119
120 # creating and assigning EIP to the ENI
121 resource "aws_eip" "eip_for_eni" {
122 domain = "vpc"
123
124 tags = {
125 Name = "EIP-For-ENI"
126 }
127
128 depends_on = [aws_internet_gateway.InternetGateWay]
129
130 lifecycle {
131 prevent_destroy = false
132 }
133 }
134
135
136 resource "aws_eip_association" "eip_attach_to_eni" {
137 network_interface_id = aws_network_interface.NetworkInterface-webserver.id
138 allocation_id = aws_eip.eip_for_eni.id
139
140 }
141
142 # create ubuntu ec2 instance and intall/enable apache2
143 resource "aws_instance" "webserver" {
144 ami = "ami-084568db4383264d4"
145 instance_type = "t2.micro"
146 availability_zone = "us-east-1a"
147 key_name = "ec2.keypair" # Replace with your key pair name
148 network_interface {
149 device_index = 0
150 network_interface_id = aws_network_interface.NetworkInterface-webserver.id
151 }
OUTLINE
TIMELINE
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform


File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
main.tf
terraform.tfstate.backup
terraform.tfstate
main.tf > resource "aws_subnet" "prod-subnet"
143 resource "aws_instance" "webserver" {
153
154 # user_data to install Apache2
155 user_data = <<-EOF
156 #!/bin/bash
157 sudo apt update -y
158 sudo apt install apache2 -y
159 sudo systemctl start apache2
160 sudo bash -c 'echo your very first web server > /var/www/html/index.html
161 EOF
162
163 lifecycle {
164 create_before_destroy = true
165 }
166
167 tags = {
168 Name = "Ubuntu-Server"
169 }
170 }
171
OUTLINE
Open a Remote Window
0 0
Ln 27, Col 38
Spaces: 2
UTF-8
CRLF
Terraform

Run ··· project-1

main.tf | terraform.tfstate.backup | terraform.tfstate

main.tf > resource "aws_subnet" "prod-subnet"

```
143  resource "aws_instance" "webserver" {
153
154    # user_data to install Apache2
155    user_data = <<-EOF
156                #!/bin/bash
157                sudo apt update -y
158                sudo apt install apache2 -y
159                sudo systemctl start apache2
160                sudo bash -c 'echo your very first web server > /var/www/html/index.html
161                EOF
162
163    lifecycle {
164      create_before_destroy = true
165    }
166
167  tags = {
168      Name = "Ubuntu-Server"
169    }
170  }
171
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

pwsh

```
PS C:\Users\TANVI\Documents\Terraform-projects\project-1> terraform plan -out=myplan.tfplan
```

Run ··· project-1

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS pwsh

```
PS C:\Users\TANVI\Documents\Terraform-projects\project-1> 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_eip.eip_for_eni will be created
  + resource "aws_eip" "eip_for_eni" {
      + allocation_id        = (known after apply)
      + arn                  = (known after apply)
      + association_id       = (known after apply)
      + carrier_ip           = (known after apply)
      + customer_owned_ip    = (known after apply)
      + domain               = "vpc"
      + id                   = (known after apply)
      + instance             = (known after apply)
      + ipam_pool_id         = (known after apply)
      + network_border_group = (known after apply)
      + network_interface    = (known after apply)
      + private_dns          = (known after apply)
      + private_ip           = (known after apply)
      + ptr_record           = (known after apply)
      + public_dns           = (known after apply)
      + public_ip            = (known after apply)
      + public_ipv4_pool     = (known after apply)
      + tags                 = {
          + "Name" = "EIP-For-ENI"
        }
      + tags_all             = {
          + "Name" = "EIP-For-ENI"
        }
      + vpc                  = (known after apply)
    }

  # aws_eip_association.eip_attach_to_eni will be created
  + resource "aws_eip_association" "eip_attach_to_eni" {
      + allocation_id        = (known after apply)
```

```
  # aws_eip_association.eip_attach_to_eni will be created
  + resource "aws_eip_association" "eip_attach_to_eni" {
      + allocation_id        = (known after apply)
      + id                   = (known after apply)
      + instance_id          = (known after apply)
      + network_interface_id = (known after apply)
      + private_ip_address   = (known after apply)
      + public_ip            = (known after apply)
    }

  # aws_instance.webserver will be created
  + resource "aws_instance" "webserver" {
      + ami                                  = "ami-084568db4383264d4"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = "us-east-1a"
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + enable_primary_ipv6                  = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = "ec2.keypair"
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
```

```
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "Ubuntu-Server"
        }
      + tags_all                             = {
          + "Name" = "Ubuntu-Server"
        }
      + tenancy                              = (known after apply)
      + user_data                            = "af45acea32513c59b78559271d36ba4e239d8314"
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)

      + capacity_reservation_specification (known after apply)

      + cpu_options (known after apply)

      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)
```

```
    + metadata_options (known after apply)

    + network_interface {
        + delete_on_termination = false
        + device_index          = 0
        + network_card_index    = 0
        + network_interface_id  = (known after apply)
      }

    + private_dns_name_options (known after apply)

    + root_block_device (known after apply)
  }

# aws_internet_gateway.InternetGateWay will be created
+ resource "aws_internet_gateway" "InternetGateWay" {
    + arn      = (known after apply)
    + id       = (known after apply)
    + owner_id = (known after apply)
    + tags     = {
        + "Name" = "MY-IGW"
      }
    + tags_all = {
        + "Name" = "MY-IGW"
      }
    + vpc_id   = (known after apply)
  }

# aws_network_interface.NetworkInterface-webserver will be created
+ resource "aws_network_interface" "NetworkInterface-webserver" {
    + arn                       = (known after apply)
    + enable_primary_ipv6       = (known after apply)
    + id                        = (known after apply)
    + interface_type            = (known after apply)
    + ipv4_prefix_count         = (known after apply)
    + ipv4_prefixes             = (known after apply)
    + ipv6_address_count        = (known after apply)
```

```
      + ipv4_prefix_count          = (known after apply)
      + ipv4_prefixes              = (known after apply)
      + ipv6_address_count         = (known after apply)
      + ipv6_address_list          = (known after apply)
      + ipv6_address_list_enabled = false
      + ipv6_addresses             = (known after apply)
      + ipv6_prefix_count          = (known after apply)
      + ipv6_prefixes              = (known after apply)
      + mac_address                = (known after apply)
      + outpost_arn                = (known after apply)
      + owner_id                   = (known after apply)
      + private_dns_name           = (known after apply)
      + private_ip                 = (known after apply)
      + private_ip_list            = (known after apply)
      + private_ip_list_enabled    = false
      + private_ips                = [
          + "10.0.1.50",
        ]
      + private_ips_count          = (known after apply)
      + security_groups            = (known after apply)
      + source_dest_check          = true
      + subnet_id                  = (known after apply)
      + tags                       = {
          + "Name" = "My-ENI"
        }
      + tags_all                   = {
          + "Name" = "My-ENI"
        }

      + attachment (known after apply)
    }

  # aws_route_table.prod-RouteTable will be created
  + resource "aws_route_table" "prod-RouteTable" {
      + arn              = (known after apply)
      + id               = (known after apply)
      + owner_id         = (known after apply)
      + propagating_vgws = (known after apply)
```

```
    + owner_id         = (known after apply)
    + propagating_vgws = (known after apply)
    + route            = [
        + {
            + cidr_block                = "0.0.0.0/0"
            + gateway_id                = (known after apply)
              # (11 unchanged attributes hidden)
          },
      ]
    + tags             = {
        + "Name" = "Production-Routetable"
      }
    + tags_all         = {
        + "Name" = "Production-Routetable"
      }
    + vpc_id           = (known after apply)
  }

# aws_route_table_association.RTAssociation will be created
+ resource "aws_route_table_association" "RTAssociation" {
    + id             = (known after apply)
    + route_table_id = (known after apply)
    + subnet_id      = (known after apply)
  }

# aws_security_group.Production-Security-Group will be created
+ resource "aws_security_group" "Production-Security-Group" {
    + arn                    = (known after apply)
    + description            = "Allow Inbound and Outbound traffic"
    + egress                 = (known after apply)
    + id                     = (known after apply)
    + ingress                = (known after apply)
    + name                   = "allow web traffic"
    + name_prefix            = (known after apply)
    + owner_id               = (known after apply)
    + revoke_rules_on_delete = false
    + tags                   = {
        + "Name" = "MY-SG"
```

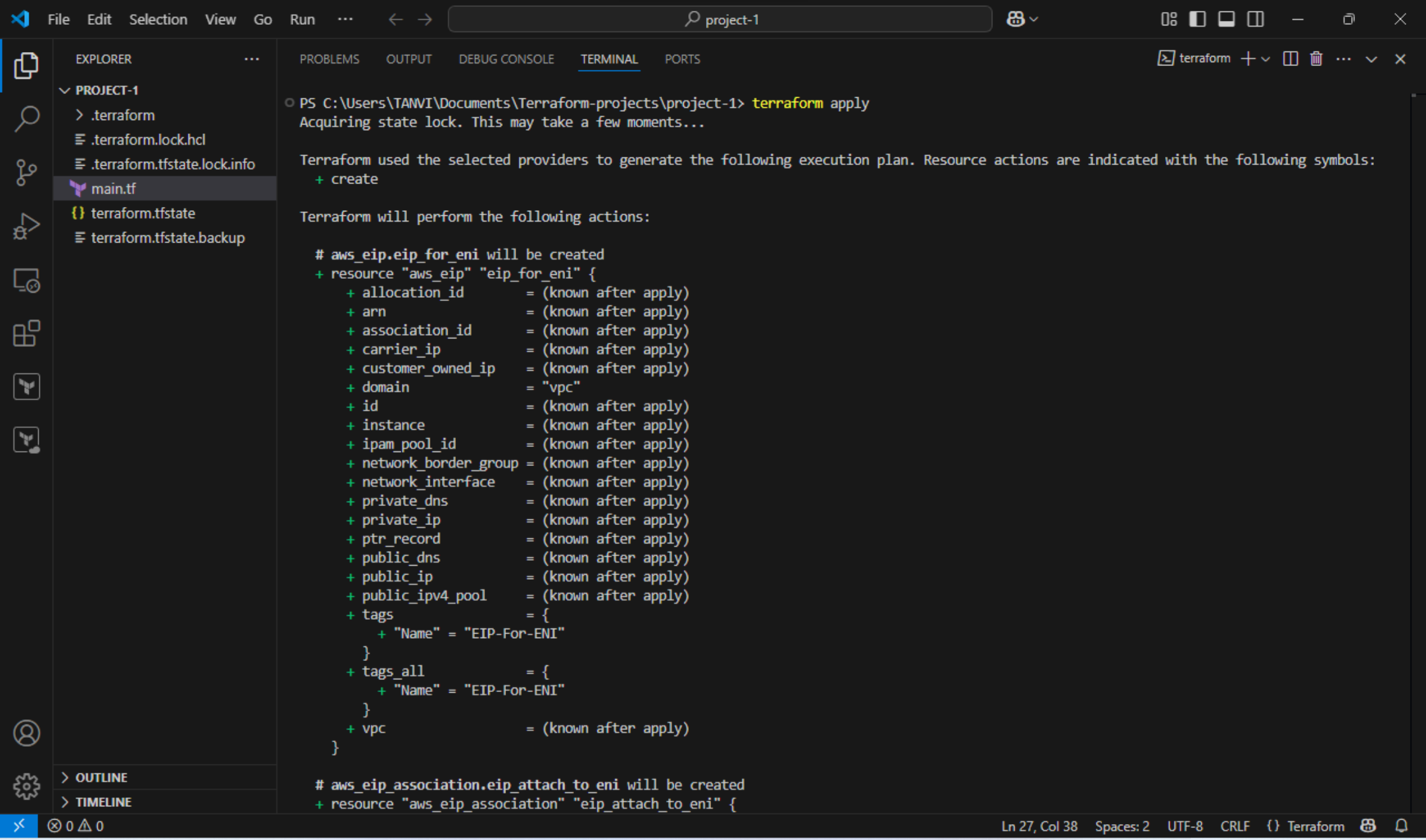
```
      + tags                  = {
          + "Name" = "MY-SG"
        }
      + tags_all              = {
          + "Name" = "MY-SG"
        }
      + vpc_id                = (known after apply)
    }

  # aws_subnet.prod-subnet will be created
  + resource "aws_subnet" "prod-subnet" {
      + arn                                            = (known after apply)
      + assign_ipv6_address_on_creation                = false
      + availability_zone                              = "us-east-1a"
      + availability_zone_id                           = (known after apply)
      + cidr_block                                     = "10.0.1.0/24"
      + enable_dns64                                   = false
      + enable_resource_name_dns_a_record_on_launch    = false
      + enable_resource_name_dns_aaaa_record_on_launch = false
      + id                                             = (known after apply)
      + ipv6_cidr_block_association_id                 = (known after apply)
      + ipv6_native                                    = false
      + map_public_ip_on_launch                        = false
      + owner_id                                       = (known after apply)
      + private_dns_hostname_type_on_launch            = (known after apply)
      + tags                                           = {
          + "Name" = "prod-subnet"
        }
      + tags_all                                       = {
          + "Name" = "prod-subnet"
        }
      + vpc_id                                         = (known after apply)
    }

  # aws_vpc.prod-vpc will be created
  + resource "aws_vpc" "prod-vpc" {
      + arn                                = (known after apply)
      + cidr_block                         = "10.0.0.0/16"
```

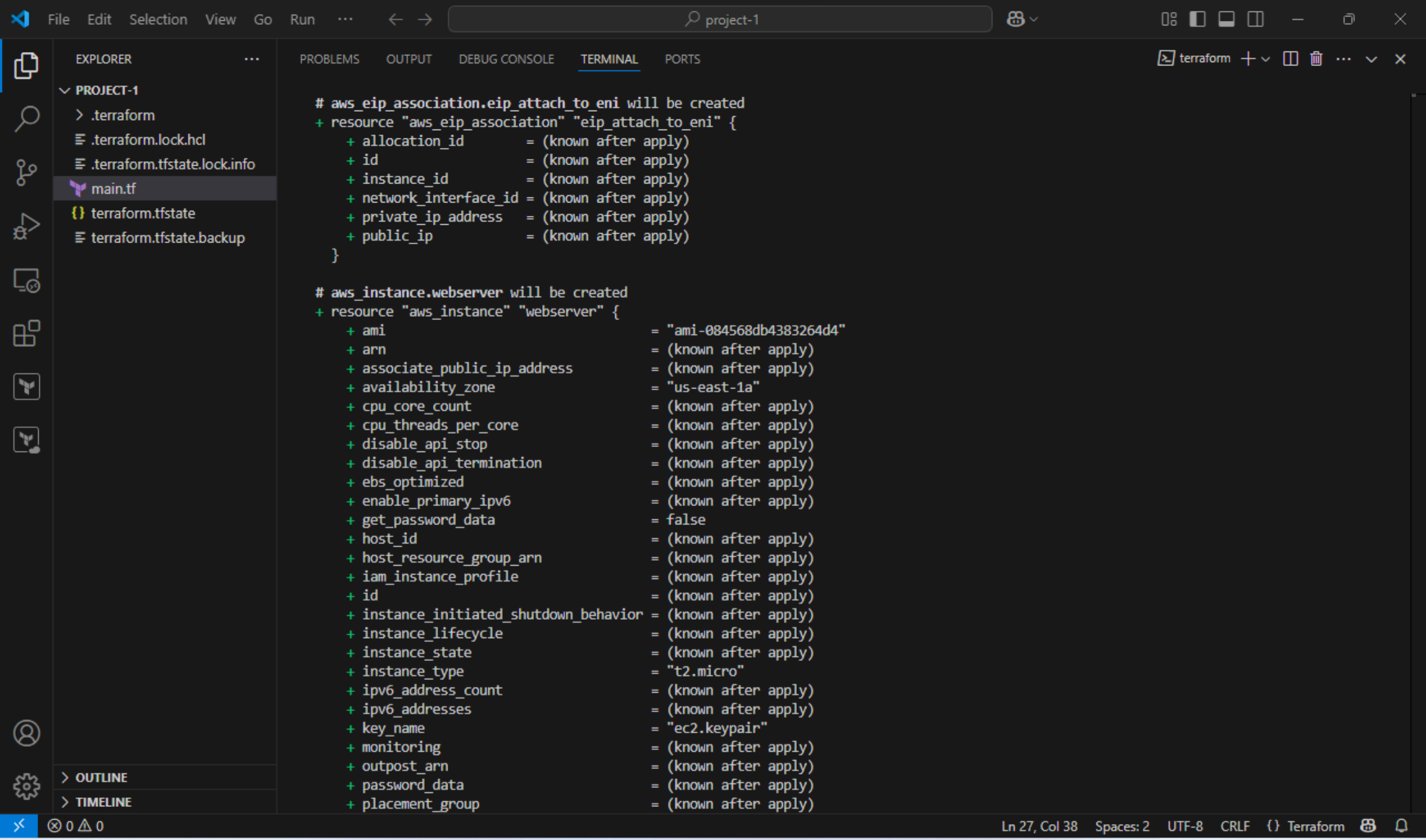
```
  # aws_vpc_security_group_ingress_rule.http will be created
  + resource "aws_vpc_security_group_ingress_rule" "http" {
      + arn                    = (known after apply)
      + cidr_ipv4              = "0.0.0.0/0"
      + description            = "HTTP"
      + from_port              = 80
      + id                     = (known after apply)
      + ip_protocol            = "tcp"
      + security_group_id      = (known after apply)
      + security_group_rule_id = (known after apply)
      + tags_all               = {}
      + to_port                = 80
    }

  # aws_vpc_security_group_ingress_rule.https will be created
  + resource "aws_vpc_security_group_ingress_rule" "https" {
      + arn                    = (known after apply)
      + cidr_ipv4              = "0.0.0.0/0"
      + description            = "HTTPS"
      + from_port              = 443
      + id                     = (known after apply)
      + ip_protocol            = "tcp"
      + security_group_id      = (known after apply)
      + security_group_rule_id = (known after apply)
      + tags_all               = {}
      + to_port                = 443
    }

  # aws_vpc_security_group_ingress_rule.ssh will be created
  + resource "aws_vpc_security_group_ingress_rule" "ssh" {
      + arn                    = (known after apply)
      + cidr_ipv4              = "0.0.0.0/0"
      + description            = "SSH"
      + from_port              = 22
      + id                     = (known after apply)
      + ip_protocol            = "tcp"
      + security_group_id      = (known after apply)
      + security_group_rule_id = (known after apply)
```

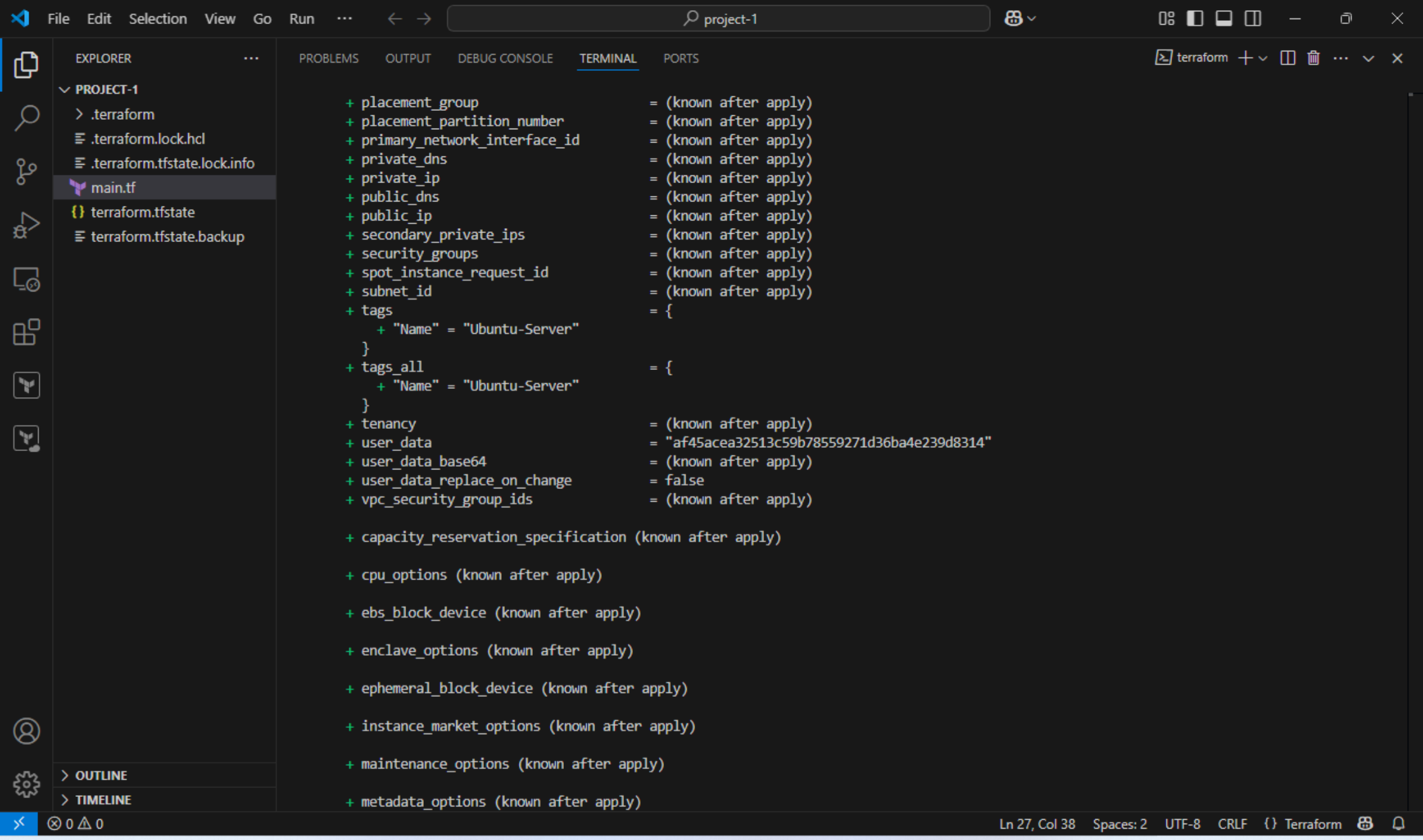
Run ··· project-1

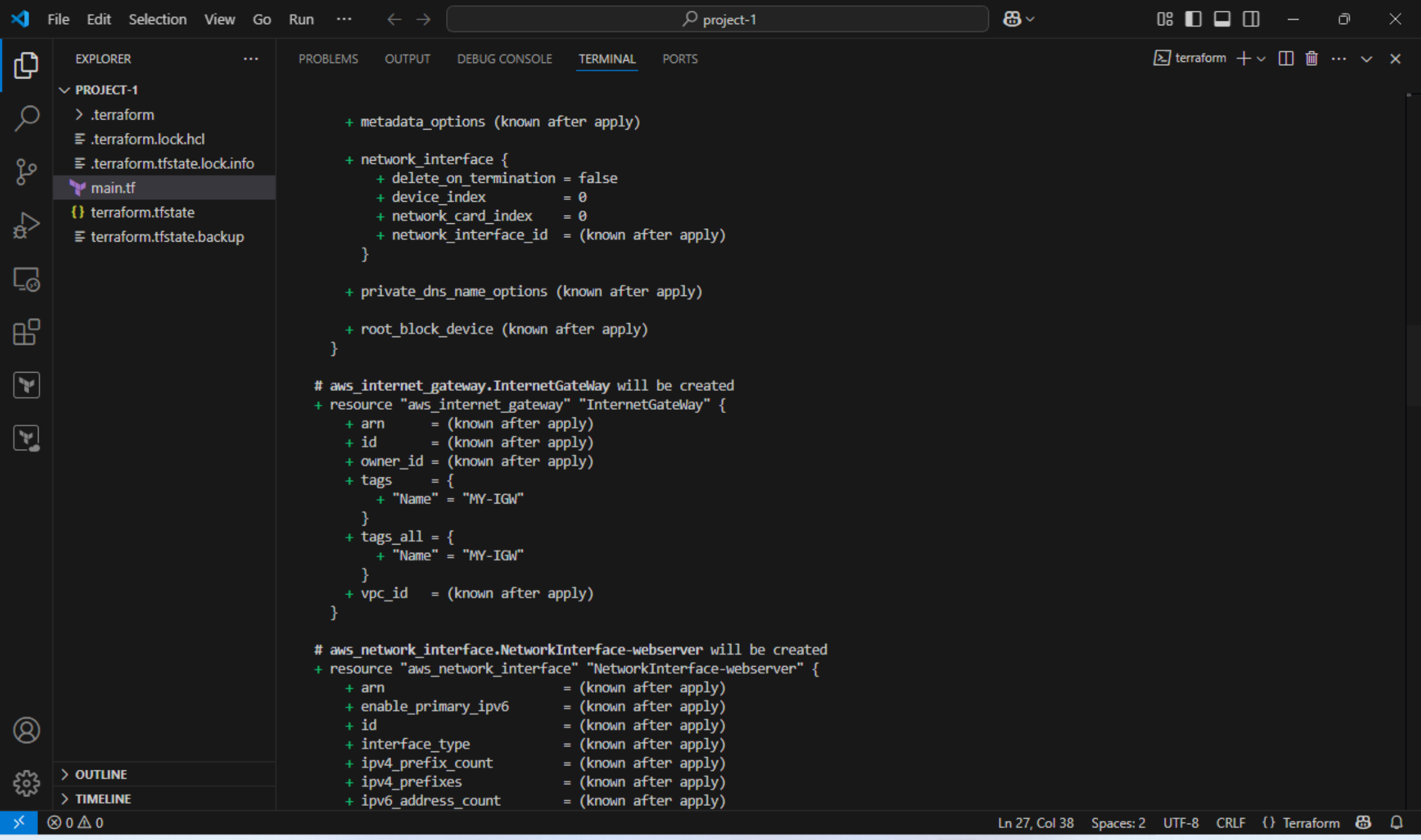
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS pwsh

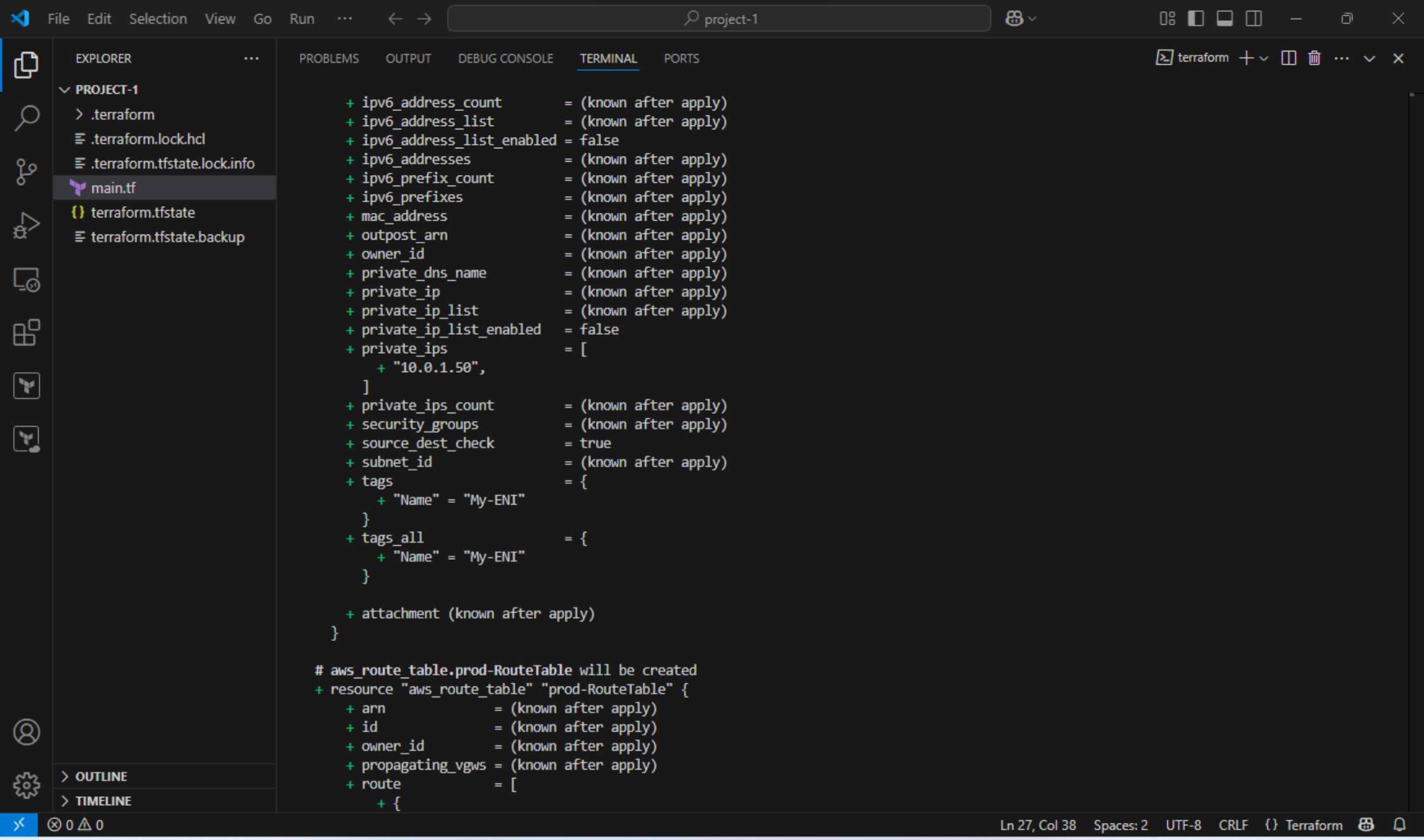
```
  # aws_vpc_security_group_ingress_rule.ssh will be created
  + resource "aws_vpc_security_group_ingress_rule" "ssh" {
      + arn                    = (known after apply)
      + cidr_ipv4              = "0.0.0.0/0"
      + description            = "SSH"
      + from_port              = 22
      + id                     = (known after apply)
      + ip_protocol            = "tcp"
      + security_group_id      = (known after apply)
      + security_group_rule_id = (known after apply)
      + tags_all               = {}
      + to_port                = 22
    }

Plan: 14 to add, 0 to change, 0 to destroy.

---

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform
apply" now.
PS C:\Users\TANVI\Documents\Terraform-projects\project-1>
```

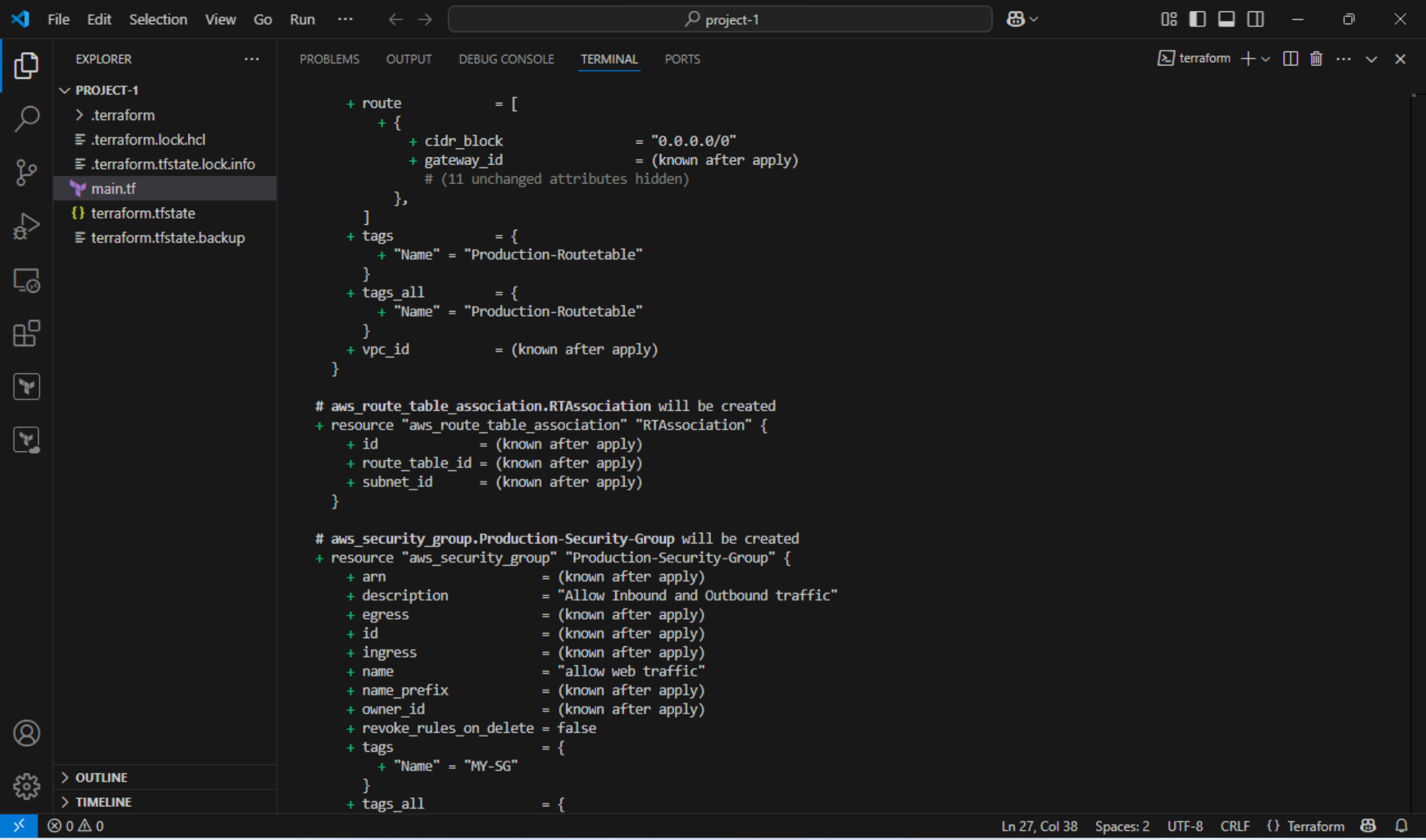

File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
PS C:\Users\TANVI\Documents\Terraform-projects\project-1> terraform apply
Acquiring state lock. This may take a few moments...
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_eip.eip_for_eni will be created
+ resource "aws_eip" "eip_for_eni" {
+ allocation_id = (known after apply)
+ arn = (known after apply)
+ association_id = (known after apply)
+ carrier_ip = (known after apply)
+ customer_owned_ip = (known after apply)
+ domain = "vpc"
+ id = (known after apply)
+ instance = (known after apply)
+ ipam_pool_id = (known after apply)
+ network_border_group = (known after apply)
+ network_interface = (known after apply)
+ private_dns = (known after apply)
+ private_ip = (known after apply)
+ ptr_record = (known after apply)
+ public_dns = (known after apply)
+ public_ip = (known after apply)
+ public_ipv4_pool = (known after apply)
+ tags = {
+ "Name" = "EIP-For-ENI"
}
+ tags_all = {
+ "Name" = "EIP-For-ENI"
}
+ vpc = (known after apply)
}
# aws_eip_association.eip_attach_to_eni will be created
+ resource "aws_eip_association" "eip_attach_to_eni" {
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF {} Terraform

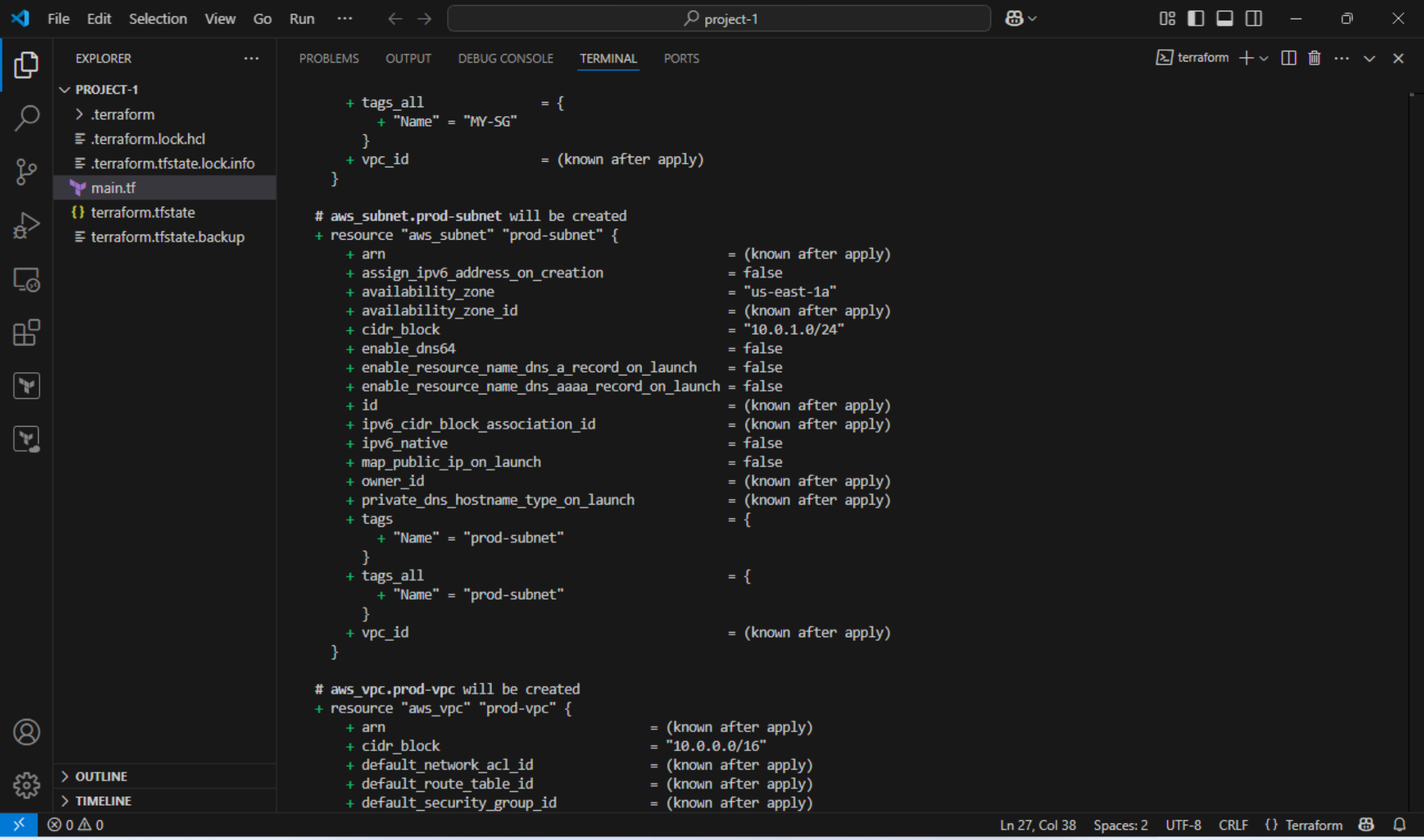

File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
# aws_eip_association.eip_attach_to_eni will be created
+ resource "aws_eip_association" "eip_attach_to_eni" {
+ allocation_id = (known after apply)
+ id = (known after apply)
+ instance_id = (known after apply)
+ network_interface_id = (known after apply)
+ private_ip_address = (known after apply)
+ public_ip = (known after apply)
}
# aws_instance.webserver will be created
+ resource "aws_instance" "webserver" {
+ ami = "ami-084568db4383264d4"
+ arn = (known after apply)
+ associate_public_ip_address = (known after apply)
+ availability_zone = "us-east-1a"
+ cpu_core_count = (known after apply)
+ cpu_threads_per_core = (known after apply)
+ disable_api_stop = (known after apply)
+ disable_api_termination = (known after apply)
+ ebs_optimized = (known after apply)
+ enable_primary_ipv6 = (known after apply)
+ get_password_data = false
+ host_id = (known after apply)
+ host_resource_group_arn = (known after apply)
+ iam_instance_profile = (known after apply)
+ id = (known after apply)
+ instance_initiated_shutdown_behavior = (known after apply)
+ instance_lifecycle = (known after apply)
+ instance_state = (known after apply)
+ instance_type = "t2.micro"
+ ipv6_address_count = (known after apply)
+ ipv6_addresses = (known after apply)
+ key_name = "ec2.keypair"
+ monitoring = (known after apply)
+ outpost_arn = (known after apply)
+ password_data = (known after apply)
+ placement_group = (known after apply)
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform

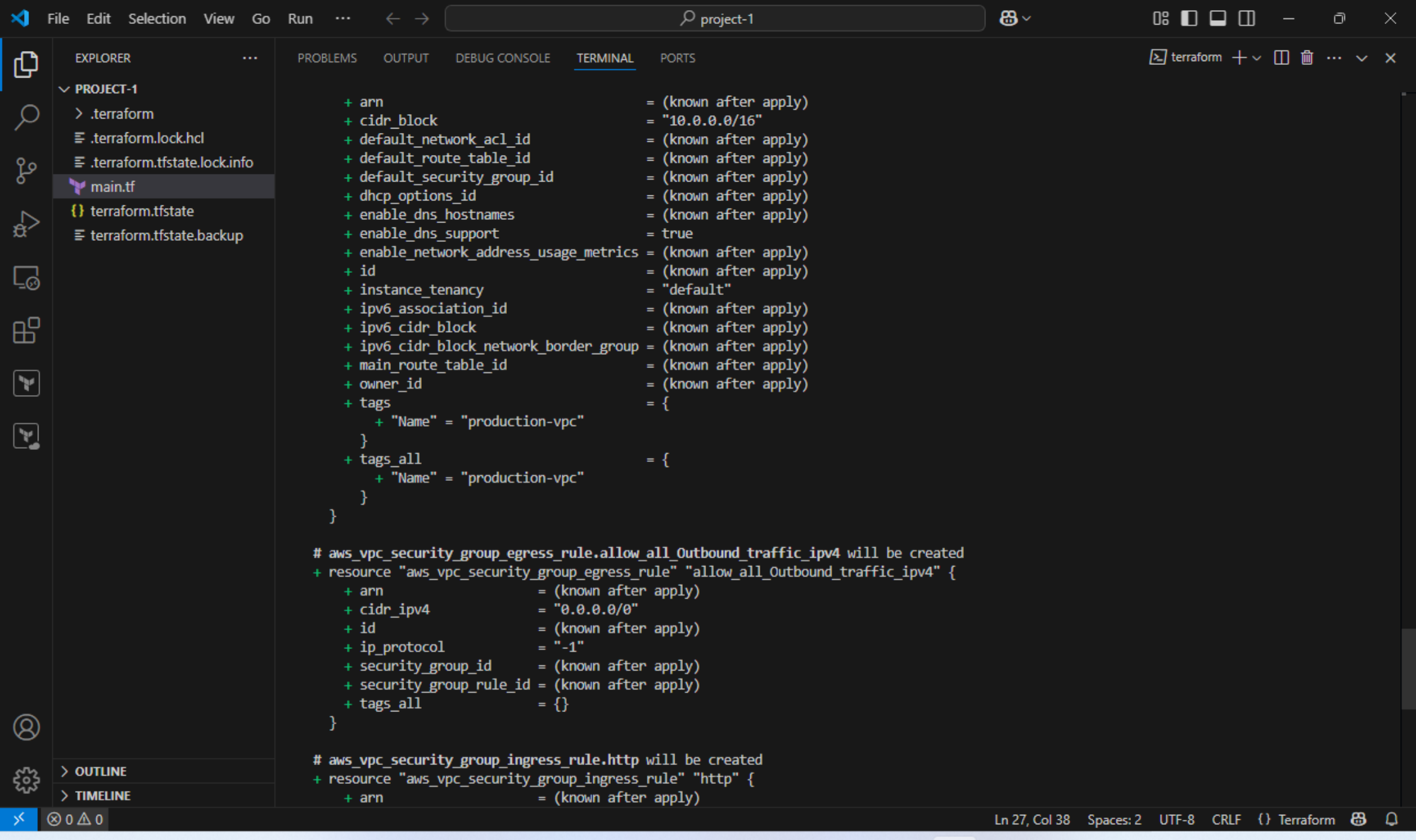

File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
+ placement_group = (known after apply)
+ placement_partition_number = (known after apply)
+ primary_network_interface_id = (known after apply)
+ private_dns = (known after apply)
+ private_ip = (known after apply)
+ public_dns = (known after apply)
+ public_ip = (known after apply)
+ secondary_private_ips = (known after apply)
+ security_groups = (known after apply)
+ spot_instance_request_id = (known after apply)
+ subnet_id = (known after apply)
+ tags = {
+ "Name" = "Ubuntu-Server"
}
+ tags_all = {
+ "Name" = "Ubuntu-Server"
}
+ tenancy = (known after apply)
+ user_data = "af45acea32513c59b78559271d36ba4e239d8314"
+ user_data_base64 = (known after apply)
+ user_data_replace_on_change = false
+ vpc_security_group_ids = (known after apply)
+ capacity_reservation_specification (known after apply)
+ cpu_options (known after apply)
+ ebs_block_device (known after apply)
+ enclave_options (known after apply)
+ ephemeral_block_device (known after apply)
+ instance_market_options (known after apply)
+ maintenance_options (known after apply)
+ metadata_options (known after apply)
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform

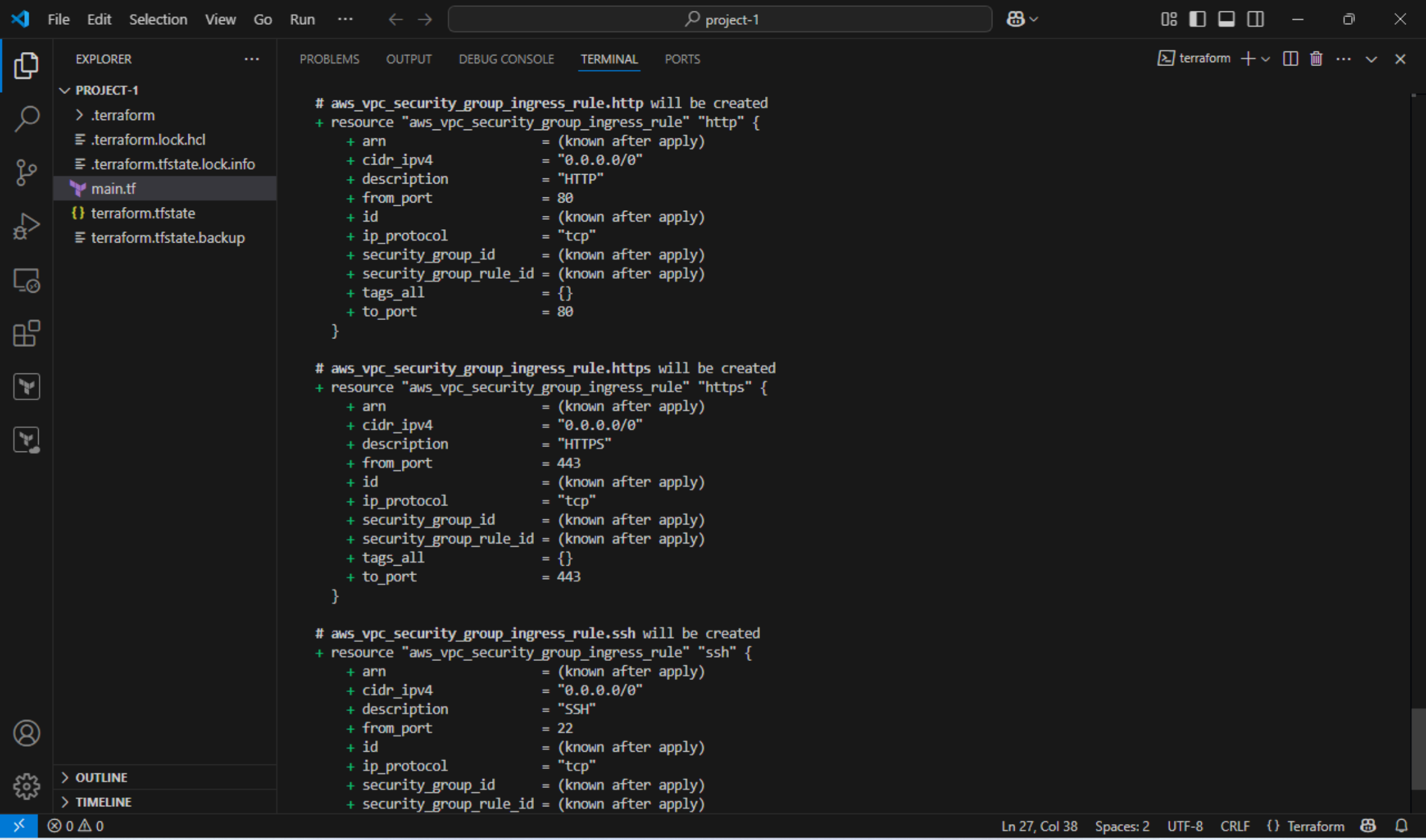

File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
+ metadata_options (known after apply)
+ network_interface {
+ delete_on_termination = false
+ device_index = 0
+ network_card_index = 0
+ network_interface_id = (known after apply)
}
+ private_dns_name_options (known after apply)
+ root_block_device (known after apply)
}
# aws_internet_gateway.InternetGateWay will be created
+ resource "aws_internet_gateway" "InternetGateWay" {
+ arn = (known after apply)
+ id = (known after apply)
+ owner_id = (known after apply)
+ tags = {
+ "Name" = "MY-IGW"
}
+ tags_all = {
+ "Name" = "MY-IGW"
}
+ vpc_id = (known after apply)
}
# aws_network_interface.NetworkInterface-webserver will be created
+ resource "aws_network_interface" "NetworkInterface-webserver" {
+ arn = (known after apply)
+ enable_primary_ipv6 = (known after apply)
+ id = (known after apply)
+ interface_type = (known after apply)
+ ipv4_prefix_count = (known after apply)
+ ipv4_prefixes = (known after apply)
+ ipv6_address_count = (known after apply)
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform

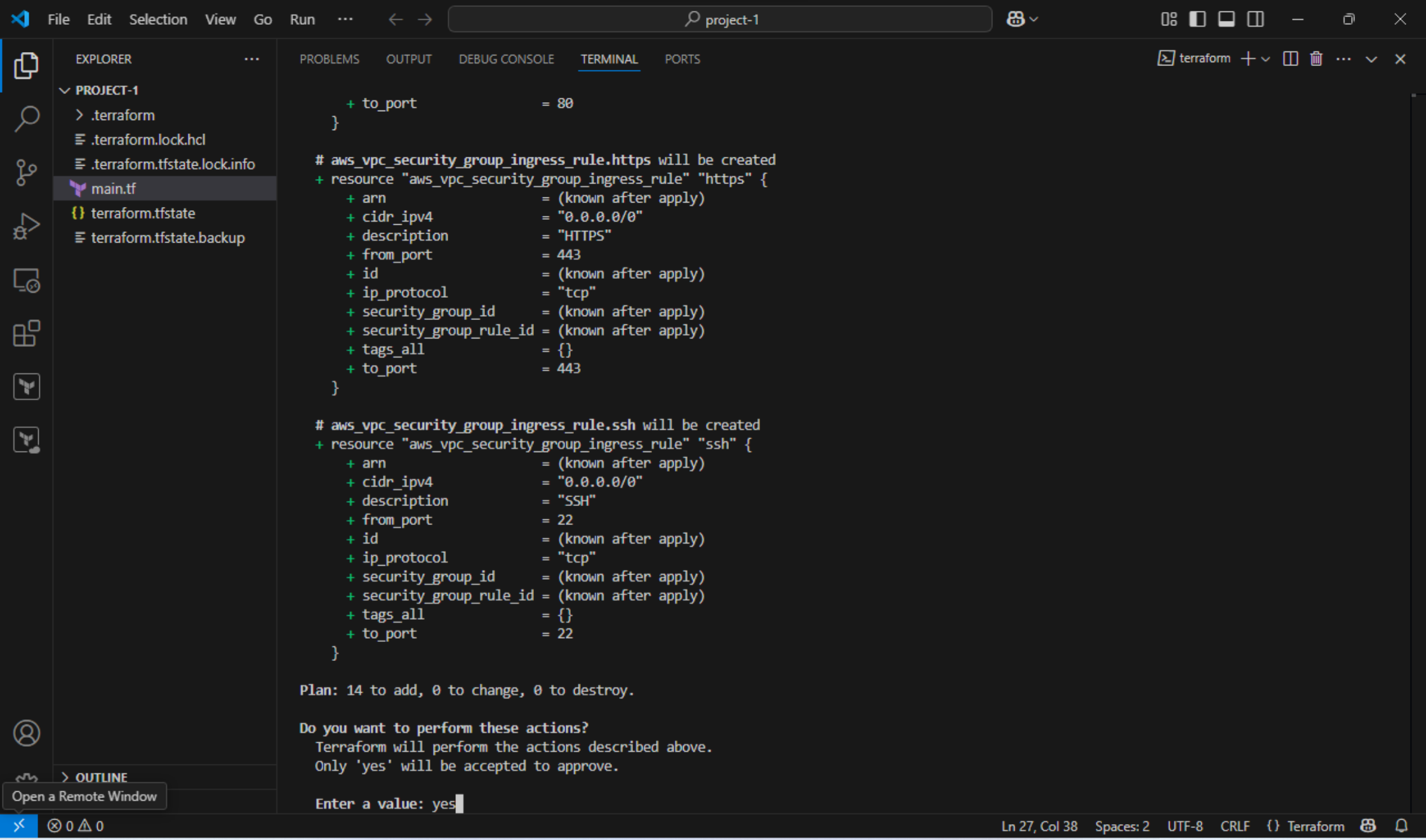

File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
+ ipv6_address_count = (known after apply)
+ ipv6_address_list = (known after apply)
+ ipv6_address_list_enabled = false
+ ipv6_addresses = (known after apply)
+ ipv6_prefix_count = (known after apply)
+ ipv6_prefixes = (known after apply)
+ mac_address = (known after apply)
+ outpost_arn = (known after apply)
+ owner_id = (known after apply)
+ private_dns_name = (known after apply)
+ private_ip = (known after apply)
+ private_ip_list = (known after apply)
+ private_ip_list_enabled = false
+ private_ips = [
+ "10.0.1.50",
]
+ private_ips_count = (known after apply)
+ security_groups = (known after apply)
+ source_dest_check = true
+ subnet_id = (known after apply)
+ tags = {
+ "Name" = "My-ENI"
}
+ tags_all = {
+ "Name" = "My-ENI"
}
+ attachment (known after apply)
}
# aws_route_table.prod-RouteTable will be created
+ resource "aws_route_table" "prod-RouteTable" {
+ arn = (known after apply)
+ id = (known after apply)
+ owner_id = (known after apply)
+ propagating_vgws = (known after apply)
+ route = [
+ {
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform

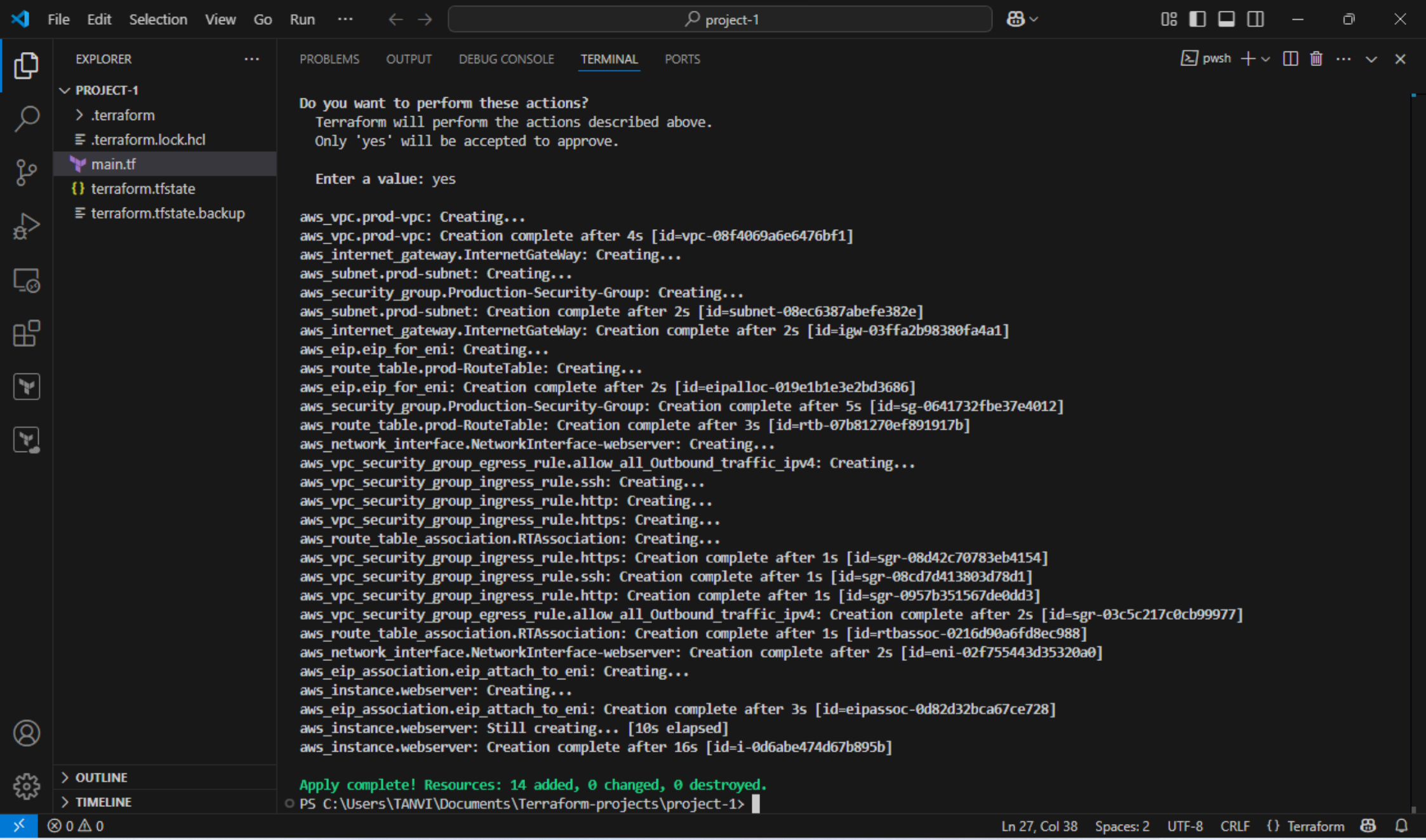

File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
+ route = [
+ {
+ cidr_block = "0.0.0.0/0"
+ gateway_id = (known after apply)
# (11 unchanged attributes hidden)
},
]
+ tags = {
+ "Name" = "Production-Routetable"
}
+ tags_all = {
+ "Name" = "Production-Routetable"
}
+ vpc_id = (known after apply)
}
# aws_route_table_association.RTAssociation will be created
+ resource "aws_route_table_association" "RTAssociation" {
+ id = (known after apply)
+ route_table_id = (known after apply)
+ subnet_id = (known after apply)
}
# aws_security_group.Production-Security-Group will be created
+ resource "aws_security_group" "Production-Security-Group" {
+ arn = (known after apply)
+ description = "Allow Inbound and Outbound traffic"
+ egress = (known after apply)
+ id = (known after apply)
+ ingress = (known after apply)
+ name = "allow web traffic"
+ name_prefix = (known after apply)
+ owner_id = (known after apply)
+ revoke_rules_on_delete = false
+ tags = {
+ "Name" = "MY-SG"
}
+ tags_all = {
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF {} Terraform

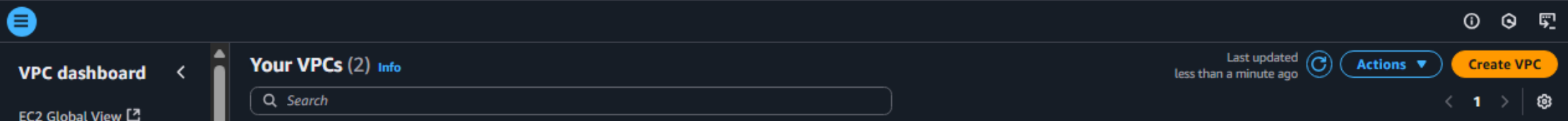

File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
+ tags_all = {
+ "Name" = "MY-SG"
}
+ vpc_id = (known after apply)
}
# aws_subnet.prod-subnet will be created
+ resource "aws_subnet" "prod-subnet" {
+ arn = (known after apply)
+ assign_ipv6_address_on_creation = false
+ availability_zone = "us-east-1a"
+ availability_zone_id = (known after apply)
+ cidr_block = "10.0.1.0/24"
+ enable_dns64 = false
+ enable_resource_name_dns_a_record_on_launch = false
+ enable_resource_name_dns_aaaa_record_on_launch = false
+ id = (known after apply)
+ ipv6_cidr_block_association_id = (known after apply)
+ ipv6_native = false
+ map_public_ip_on_launch = false
+ owner_id = (known after apply)
+ private_dns_hostname_type_on_launch = (known after apply)
+ tags = {
+ "Name" = "prod-subnet"
}
+ tags_all = {
+ "Name" = "prod-subnet"
}
+ vpc_id = (known after apply)
}
# aws_vpc.prod-vpc will be created
+ resource "aws_vpc" "prod-vpc" {
+ arn = (known after apply)
+ cidr_block = "10.0.0.0/16"
+ default_network_acl_id = (known after apply)
+ default_route_table_id = (known after apply)
+ default_security_group_id = (known after apply)
⊗ 0 ⚠ 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF {} Terraform

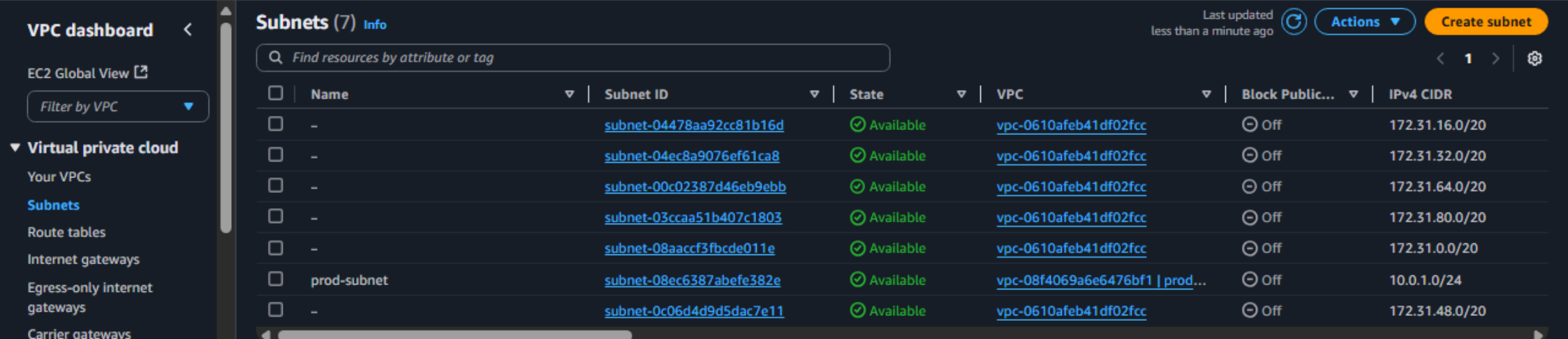

File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
+ arn = (known after apply)
+ cidr_block = "10.0.0.0/16"
+ default_network_acl_id = (known after apply)
+ default_route_table_id = (known after apply)
+ default_security_group_id = (known after apply)
+ dhcp_options_id = (known after apply)
+ enable_dns_hostnames = (known after apply)
+ enable_dns_support = true
+ enable_network_address_usage_metrics = (known after apply)
+ id = (known after apply)
+ instance_tenancy = "default"
+ ipv6_association_id = (known after apply)
+ ipv6_cidr_block = (known after apply)
+ ipv6_cidr_block_network_border_group = (known after apply)
+ main_route_table_id = (known after apply)
+ owner_id = (known after apply)
+ tags = {
+ "Name" = "production-vpc"
}
+ tags_all = {
+ "Name" = "production-vpc"
}
}
# aws_vpc_security_group_egress_rule.allow_all_Outbound_traffic_ipv4 will be created
+ resource "aws_vpc_security_group_egress_rule" "allow_all_Outbound_traffic_ipv4" {
+ arn = (known after apply)
+ cidr_ipv4 = "0.0.0.0/0"
+ id = (known after apply)
+ ip_protocol = "-1"
+ security_group_id = (known after apply)
+ security_group_rule_id = (known after apply)
+ tags_all = {}
}
# aws_vpc_security_group_ingress_rule.http will be created
+ resource "aws_vpc_security_group_ingress_rule" "http" {
+ arn = (known after apply)
⊗ 0 ⚠ 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF {} Terraform


File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
# aws_vpc_security_group_ingress_rule.http will be created
+ resource "aws_vpc_security_group_ingress_rule" "http" {
+ arn = (known after apply)
+ cidr_ipv4 = "0.0.0.0/0"
+ description = "HTTP"
+ from_port = 80
+ id = (known after apply)
+ ip_protocol = "tcp"
+ security_group_id = (known after apply)
+ security_group_rule_id = (known after apply)
+ tags_all = {}
+ to_port = 80
}
# aws_vpc_security_group_ingress_rule.https will be created
+ resource "aws_vpc_security_group_ingress_rule" "https" {
+ arn = (known after apply)
+ cidr_ipv4 = "0.0.0.0/0"
+ description = "HTTPS"
+ from_port = 443
+ id = (known after apply)
+ ip_protocol = "tcp"
+ security_group_id = (known after apply)
+ security_group_rule_id = (known after apply)
+ tags_all = {}
+ to_port = 443
}
# aws_vpc_security_group_ingress_rule.ssh will be created
+ resource "aws_vpc_security_group_ingress_rule" "ssh" {
+ arn = (known after apply)
+ cidr_ipv4 = "0.0.0.0/0"
+ description = "SSH"
+ from_port = 22
+ id = (known after apply)
+ ip_protocol = "tcp"
+ security_group_id = (known after apply)
+ security_group_rule_id = (known after apply)
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform


File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
Open a Remote Window
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
terraform
+ to_port = 80
}
# aws_vpc_security_group_ingress_rule.https will be created
+ resource "aws_vpc_security_group_ingress_rule" "https" {
+ arn = (known after apply)
+ cidr_ipv4 = "0.0.0.0/0"
+ description = "HTTPS"
+ from_port = 443
+ id = (known after apply)
+ ip_protocol = "tcp"
+ security_group_id = (known after apply)
+ security_group_rule_id = (known after apply)
+ tags_all = {}
+ to_port = 443
}
# aws_vpc_security_group_ingress_rule.ssh will be created
+ resource "aws_vpc_security_group_ingress_rule" "ssh" {
+ arn = (known after apply)
+ cidr_ipv4 = "0.0.0.0/0"
+ description = "SSH"
+ from_port = 22
+ id = (known after apply)
+ ip_protocol = "tcp"
+ security_group_id = (known after apply)
+ security_group_rule_id = (known after apply)
+ tags_all = {}
+ to_port = 22
}
Plan: 14 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
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform


File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
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_vpc.prod-vpc: Creating...
aws_vpc.prod-vpc: Creation complete after 4s [id=vpc-08f4069a6e6476bf1]
aws_internet_gateway.InternetGateWay: Creating...
aws_subnet.prod-subnet: Creating...
aws_security_group.Production-Security-Group: Creating...
aws_subnet.prod-subnet: Creation complete after 2s [id=subnet-08ec6387abefe382e]
aws_internet_gateway.InternetGateWay: Creation complete after 2s [id=igw-03ffa2b98380fa4a1]
aws_eip.eip_for_eni: Creating...
aws_route_table.prod-RouteTable: Creating...
aws_eip.eip_for_eni: Creation complete after 2s [id=eipalloc-019e1b1e3e2bd3686]
aws_security_group.Production-Security-Group: Creation complete after 5s [id=sg-0641732fbe37e4012]
aws_route_table.prod-RouteTable: Creation complete after 3s [id=rtb-07b81270ef891917b]
aws_network_interface.NetworkInterface-webserver: Creating...
aws_vpc_security_group_egress_rule.allow_all_Outbound_traffic_ipv4: Creating...
aws_vpc_security_group_ingress_rule.ssh: Creating...
aws_vpc_security_group_ingress_rule.http: Creating...
aws_vpc_security_group_ingress_rule.https: Creating...
aws_route_table_association.RTAssociation: Creating...
aws_vpc_security_group_ingress_rule.https: Creation complete after 1s [id=sgr-08d42c70783eb4154]
aws_vpc_security_group_ingress_rule.ssh: Creation complete after 1s [id=sgr-08cd7d413803d78d1]
aws_vpc_security_group_ingress_rule.http: Creation complete after 1s [id=sgr-0957b351567de0dd3]
aws_vpc_security_group_egress_rule.allow_all_Outbound_traffic_ipv4: Creation complete after 2s [id=sgr-03c5c217c0cb99977]
aws_route_table_association.RTAssociation: Creation complete after 1s [id=rtbassoc-0216d90a6fd8ec988]
aws_network_interface.NetworkInterface-webserver: Creation complete after 2s [id=eni-02f755443d35320a0]
aws_eip_association.eip_attach_to_eni: Creating...
aws_instance.webserver: Creating...
aws_eip_association.eip_attach_to_eni: Creation complete after 3s [id=eipassoc-0d82d32bca67ce728]
aws_instance.webserver: Still creating... [10s elapsed]
aws_instance.webserver: Creation complete after 16s [id=i-0d6abe474d67b895b]
Apply complete! Resources: 14 added, 0 changed, 0 destroyed.
PS C:\Users\TANVI\Documents\Terraform-projects\project-1>
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform


VPC dashboard
EC2 Global View
Filter by VPC
Virtual private cloud
Your VPCs
Your VPCs (2) Info
Last updated
less than a minute ago
Actions
Create VPC
Search
1
Name
VPC ID
State
Block Public...
IPv4 CIDR
IPv6 CIDR
–
vpc-0610afeb41df02fcc
Available
Off
172.31.0.0/16
–
production-vpc
vpc-08f4069a6e6476bf1
Available
Off
10.0.0.0/16
–


VPC dashboard
EC2 Global View
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
Carrier gateways
Subnets (7) Info
Last updated less than a minute ago
Actions
Create subnet
Find resources by attribute or tag
1
Name
Subnet ID
State
VPC
Block Public...
IPv4 CIDR
- subnet-04478aa92cc81b16d Available vpc-0610afeb41df02fcc Off 172.31.16.0/20
- subnet-04ec8a9076ef61ca8 Available vpc-0610afeb41df02fcc Off 172.31.32.0/20
- subnet-00c02387d46eb9ebb Available vpc-0610afeb41df02fcc Off 172.31.64.0/20
- subnet-03ccaa51b407c1803 Available vpc-0610afeb41df02fcc Off 172.31.80.0/20
- subnet-08aaccf3fbcde011e Available vpc-0610afeb41df02fcc Off 172.31.0.0/20
prod-subnet subnet-08ec6387abefe382e Available vpc-08f4069a6e6476bf1 | prod... Off 10.0.1.0/24
- subnet-0c06d4d9d5dac7e11 Available vpc-0610afeb41df02fcc Off 172.31.48.0/20

Route tables (1/3) Info

Last updated 1 minute ago | Actions | Create route table

Find resources by attribute or tag

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	O
–	rtb-0b756a4f084678fd8	–	–	Yes	vpc-0610afeb41df02fcc	72
–	rtb-0accf2c86ae2296bc	–	–	Yes	vpc-08f4069a6e6476bf1 | prod...	72
Production-Routetable	rtb-07b81270ef891917b	subnet-08ec6387abefe38...	–	No	vpc-08f4069a6e6476bf1 | prod...	72

rtb-07b81270ef891917b / Production-Routetable

Details | Routes | Subnet associations | Edge associations | Route propagation | Tags

Details

Route table ID
rtb-07b81270ef891917b

Main
No

Explicit subnet associations
subnet-08ec6387abefe382e / prod-subnet

Edge associations
–

VPC
vpc-08f4069a6e6476bf1 | production-vpc

Owner ID
720353140281

Route tables (1/3) Info

Last updated 1 minute ago | Actions | Create route table

Find resources by attribute or tag

1

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	O
-	rtb-0b756a4f084678fd8	-	-	Yes	vpc-0610afeb41df02fcc	72
-	rtb-0accf2c86ae2296bc	-	-	Yes	vpc-08f4069a6e6476bf1 | prod...	72
Production-Routetable	rtb-07b81270ef891917b	subnet-08ec6387abefe38...	-	No	vpc-08f4069a6e6476bf1 | prod...	72

rtb-07b81270ef891917b / Production-Routetable

Details | Routes | Subnet associations | Edge associations | Route propagation | Tags

Routes (2)

Both | Edit routes

Filter routes

1

Destination	Target	Status	Propagated
0.0.0.0/0	igw-03ffa2b98380fa4a1	Active	No
10.0.0.0/16	local	Active	No

Route tables (1/3) Info

Last updated 1 minute ago | Actions | Create route table

Find resources by attribute or tag

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	O
-	rtb-0b756a4f084678fd8	-	-	Yes	vpc-0610afeb41df02fcc	72
-	rtb-0accf2c86ae2296bc	-	-	Yes	vpc-08f4069a6e6476bf1 | prod...	72
Production-Routetable	rtb-07b81270ef891917b	subnet-08ec6387abefe38...	-	No	vpc-08f4069a6e6476bf1 | prod...	72

rtb-07b81270ef891917b / Production-Routetable

Details | Routes | Subnet associations | Edge associations | Route propagation | Tags

Explicit subnet associations (1)

Edit subnet associations

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
prod-subnet	subnet-08ec6387abefe382e	10.0.1.0/24	-

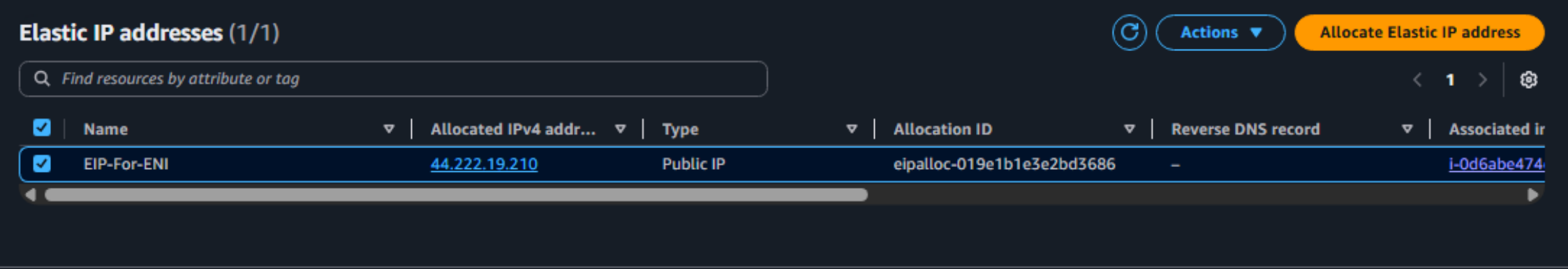
Internet gateways (2) Info

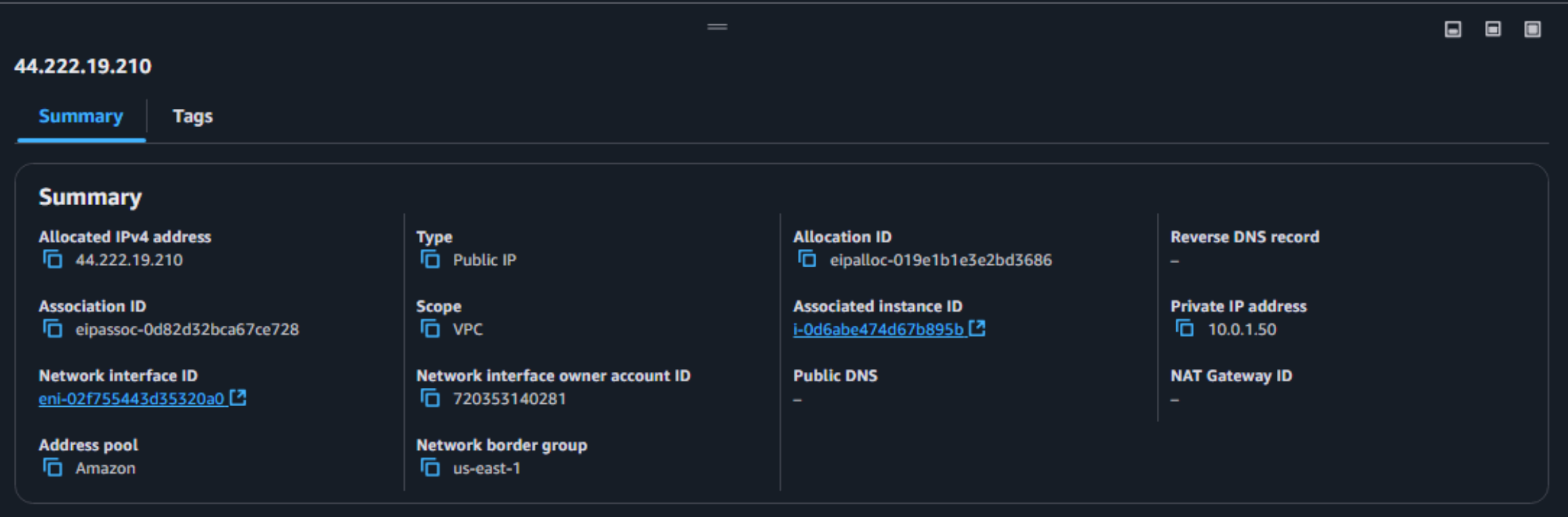
Actions ▼ | Create internet gateway

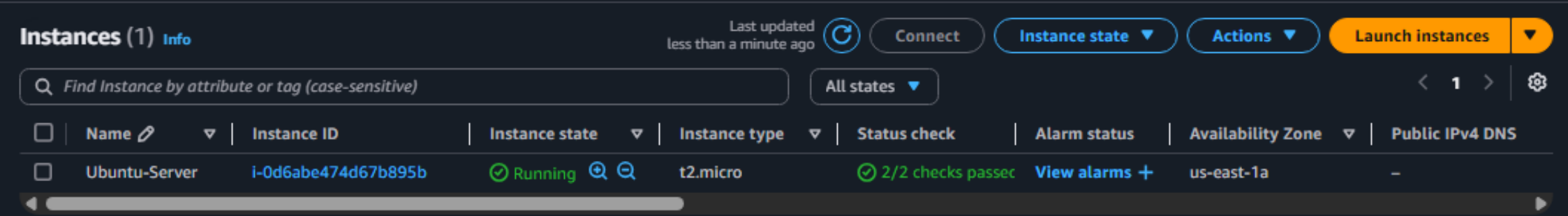
Search

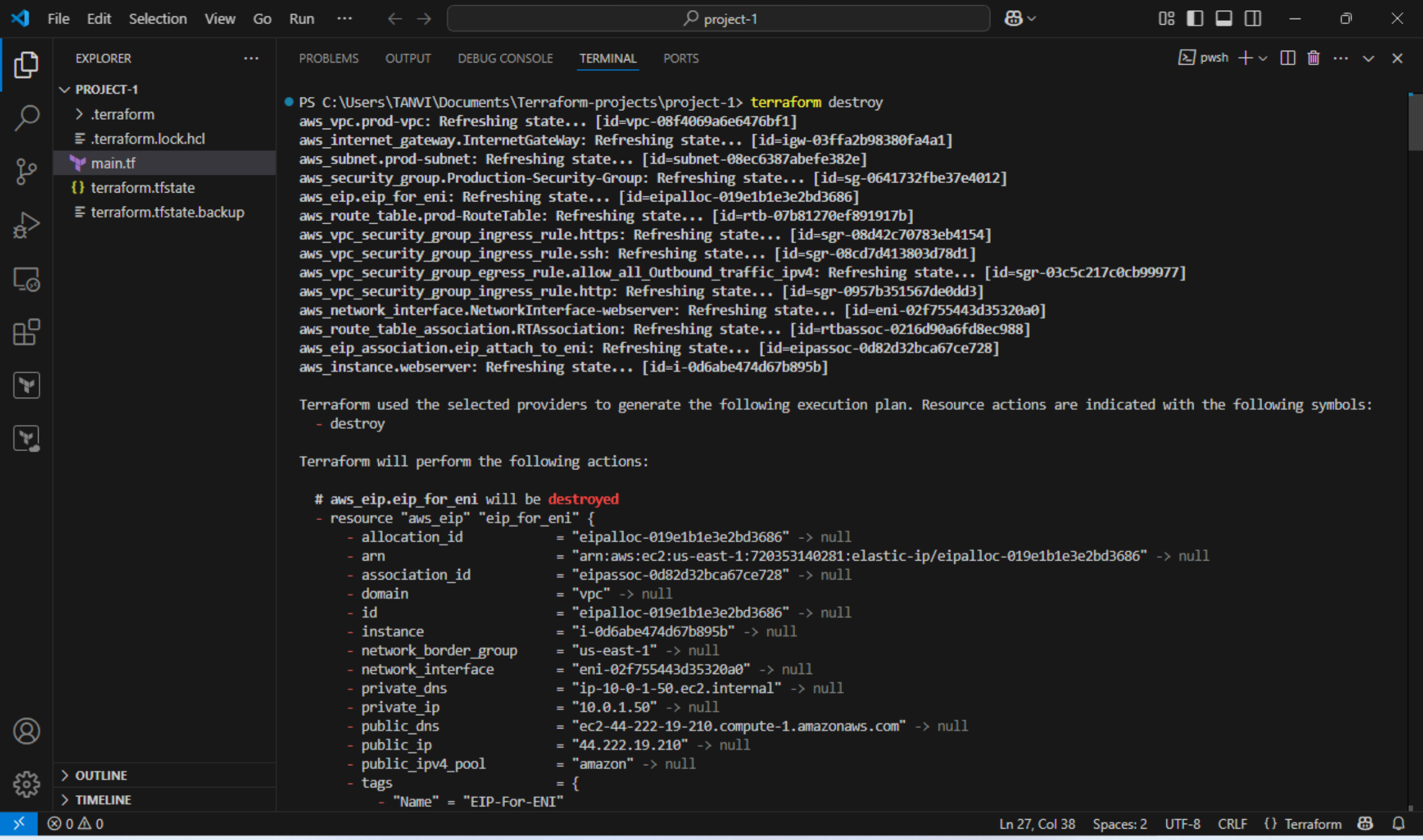
1

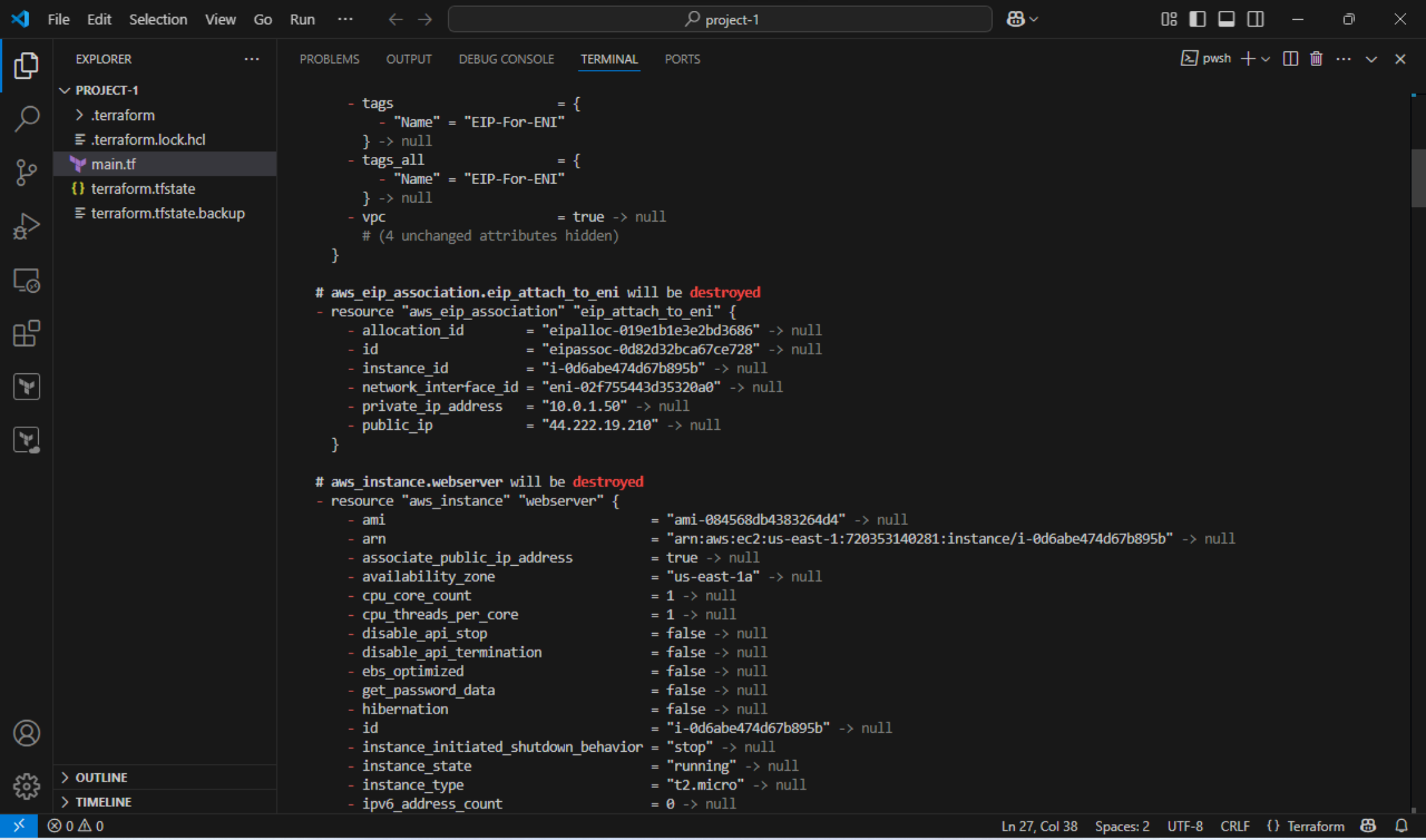
Name	Internet gateway ID	State	VPC ID	Owner
-	igw-0a4939153b2d551f2	Attached	vpc-0610afeb41df02fcc	720353140281
MY-IGW	igw-03ffa2b98380fa4a1	Attached	vpc-08f4069a6e6476bf1 | production-vpc	720353140281

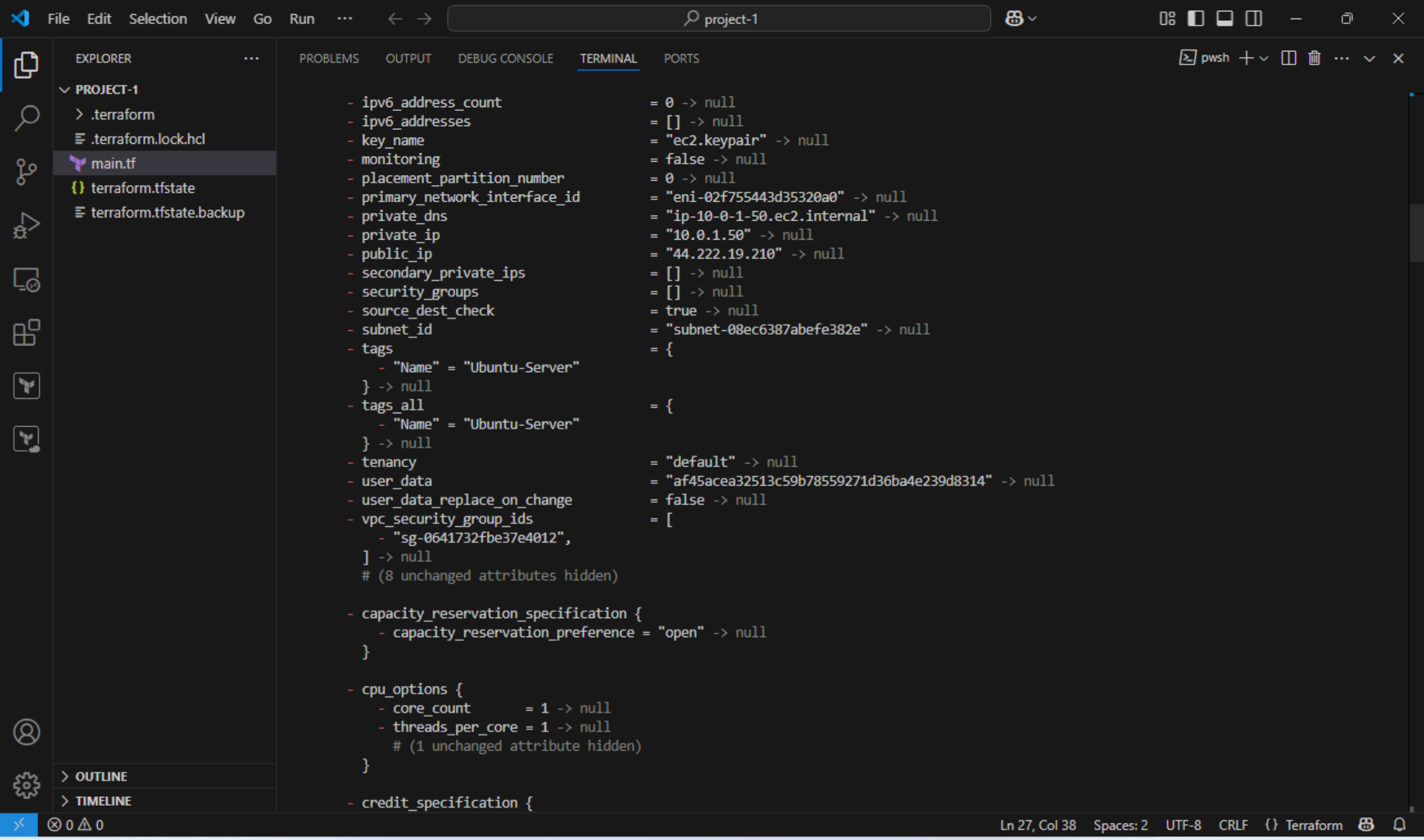

Elastic IP addresses (1/1)
Actions
Allocate Elastic IP address
Find resources by attribute or tag
1
Name
Allocated IPv4 addr...
Type
Allocation ID
Reverse DNS record
Associated ir
EIP-For-ENI
44.222.19.210
Public IP
eipalloc-019e1b1e3e2bd3686
–
i-0d6abe474

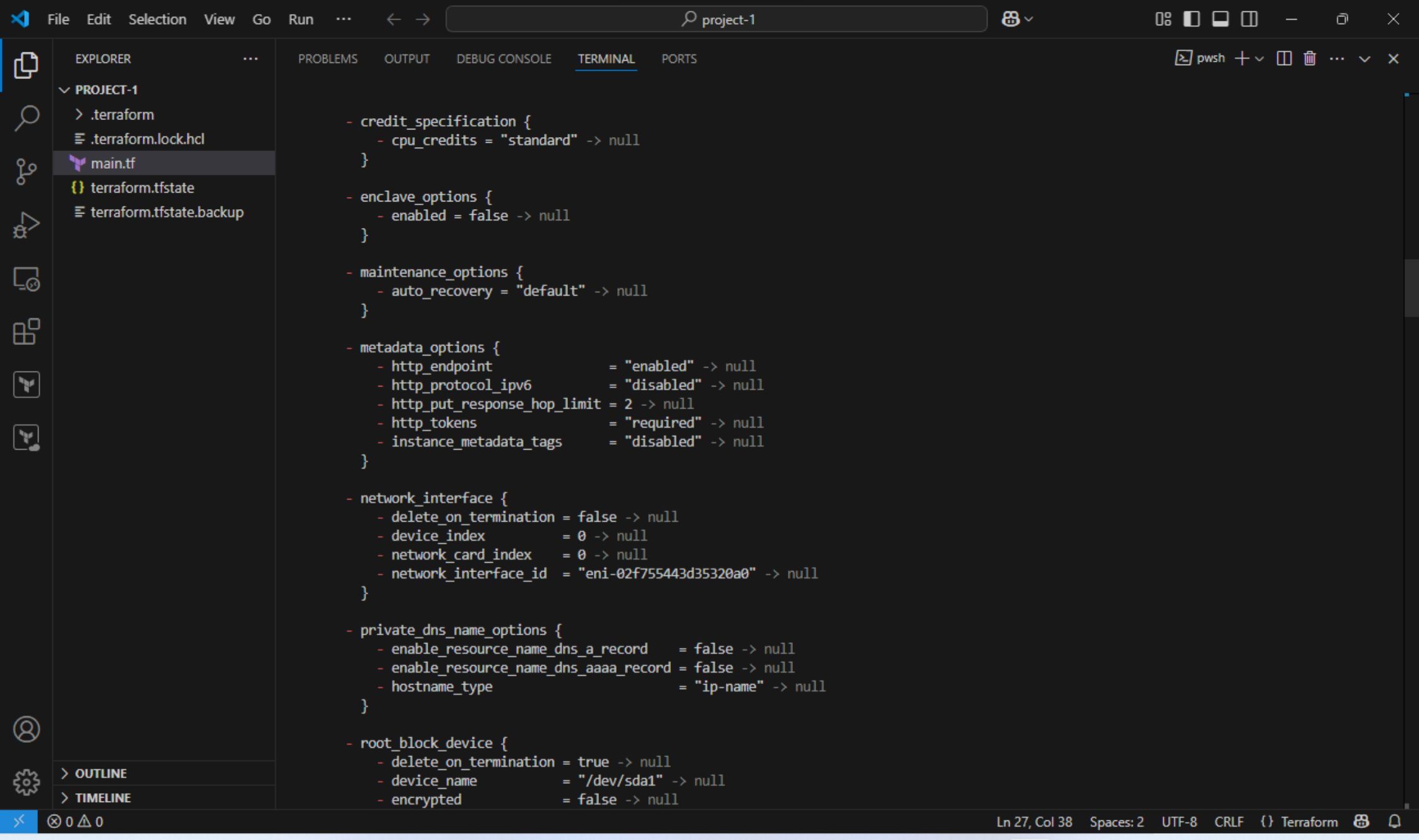

44.222.19.210
Summary
Tags
Summary
Allocated IPv4 address
44.222.19.210
Type
Public IP
Allocation ID
eipalloc-019e1b1e3e2bd3686
Reverse DNS record
–
Association ID
eipassoc-0d82d32bca67ce728
Scope
VPC
Associated instance ID
i-0d6abe474d67b895b
Private IP address
10.0.1.50
Network interface ID
eni-02f755443d35320a0
Network interface owner account ID
720353140281
Public DNS
–
NAT Gateway ID
–
Address pool
Amazon
Network border group
us-east-1

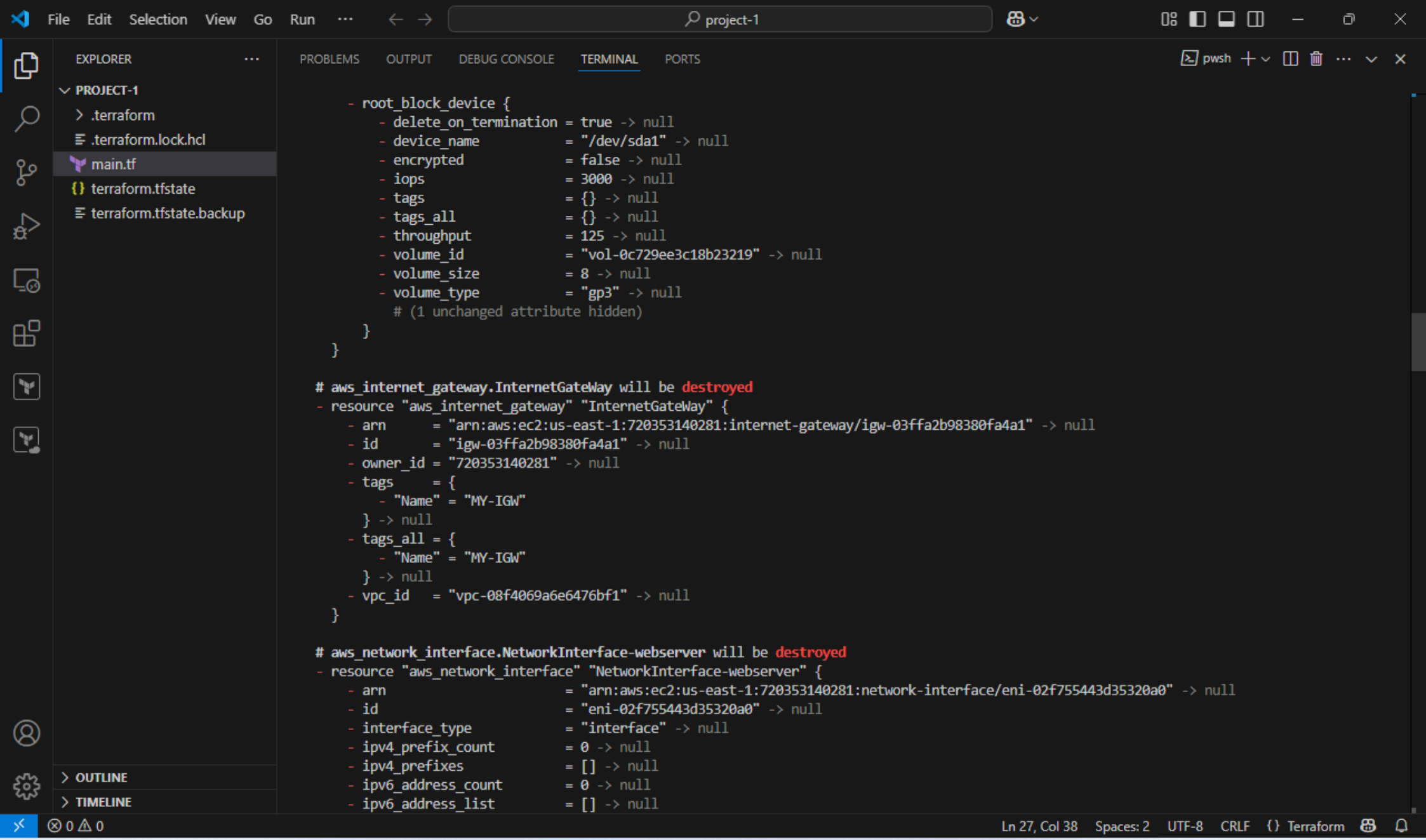

Instances (1) Info
Last updated
less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4 DNS
Ubuntu-Server
i-0d6abe474d67b895b
Running
t2.micro
2/2 checks passec
View alarms
us-east-1a
–

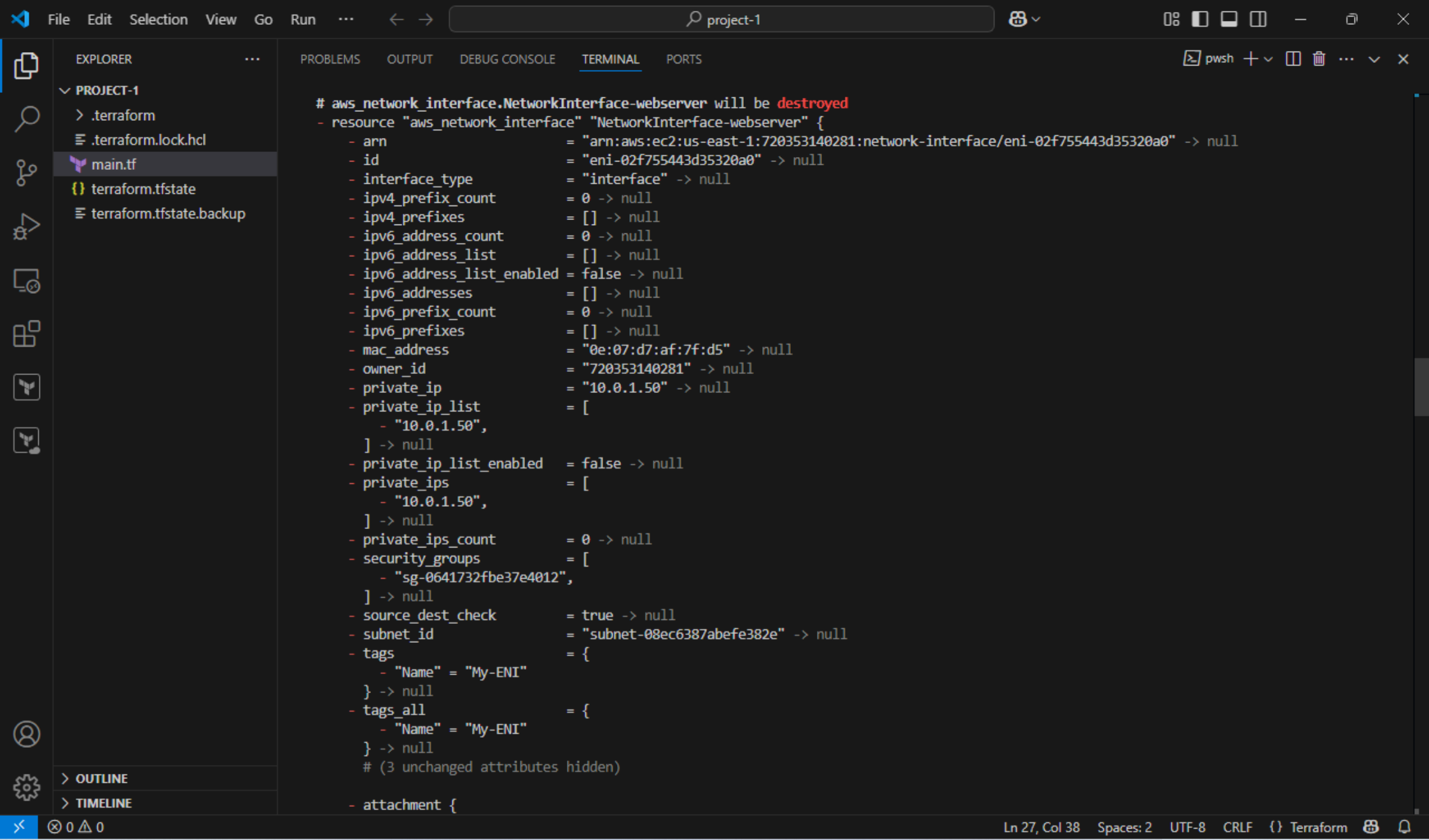

File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
PS C:\Users\TANVI\Documents\Terraform-projects\project-1> terraform destroy
aws_vpc.prod-vpc: Refreshing state... [id=vpc-08f4069a6e6476bf1]
aws_internet_gateway.InternetGateWay: Refreshing state... [id=igw-03ffa2b98380fa4a1]
aws_subnet.prod-subnet: Refreshing state... [id=subnet-08ec6387abefe382e]
aws_security_group.Production-Security-Group: Refreshing state... [id=sg-0641732fbe37e4012]
aws_eip.eip_for_eni: Refreshing state... [id=eipalloc-019e1b1e3e2bd3686]
aws_route_table.prod-RouteTable: Refreshing state... [id=rtb-07b81270ef891917b]
aws_vpc_security_group_ingress_rule.https: Refreshing state... [id=sgr-08d42c70783eb4154]
aws_vpc_security_group_ingress_rule.ssh: Refreshing state... [id=sgr-08cd7d413803d78d1]
aws_vpc_security_group_egress_rule.allow_all_Outbound_traffic_ipv4: Refreshing state... [id=sgr-03c5c217c0cb99977]
aws_vpc_security_group_ingress_rule.http: Refreshing state... [id=sgr-0957b351567de0dd3]
aws_network_interface.NetworkInterface-webserver: Refreshing state... [id=eni-02f755443d35320a0]
aws_route_table_association.RTAssociation: Refreshing state... [id=rtbassoc-0216d90a6fd8ec988]
aws_eip_association.eip_attach_to_eni: Refreshing state... [id=eipassoc-0d82d32bca67ce728]
aws_instance.webserver: Refreshing state... [id=i-0d6abe474d67b895b]
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
- destroy
Terraform will perform the following actions:
# aws_eip.eip_for_eni will be destroyed
- resource "aws_eip" "eip_for_eni" {
- allocation_id = "eipalloc-019e1b1e3e2bd3686" -> null
- arn = "arn:aws:ec2:us-east-1:720353140281:elastic-ip/eipalloc-019e1b1e3e2bd3686" -> null
- association_id = "eipassoc-0d82d32bca67ce728" -> null
- domain = "vpc" -> null
- id = "eipalloc-019e1b1e3e2bd3686" -> null
- instance = "i-0d6abe474d67b895b" -> null
- network_border_group = "us-east-1" -> null
- network_interface = "eni-02f755443d35320a0" -> null
- private_dns = "ip-10-0-1-50.ec2.internal" -> null
- private_ip = "10.0.1.50" -> null
- public_dns = "ec2-44-222-19-210.compute-1.amazonaws.com" -> null
- public_ip = "44.222.19.210" -> null
- public_ipv4_pool = "amazon" -> null
- tags = {
- "Name" = "EIP-For-ENI"
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform

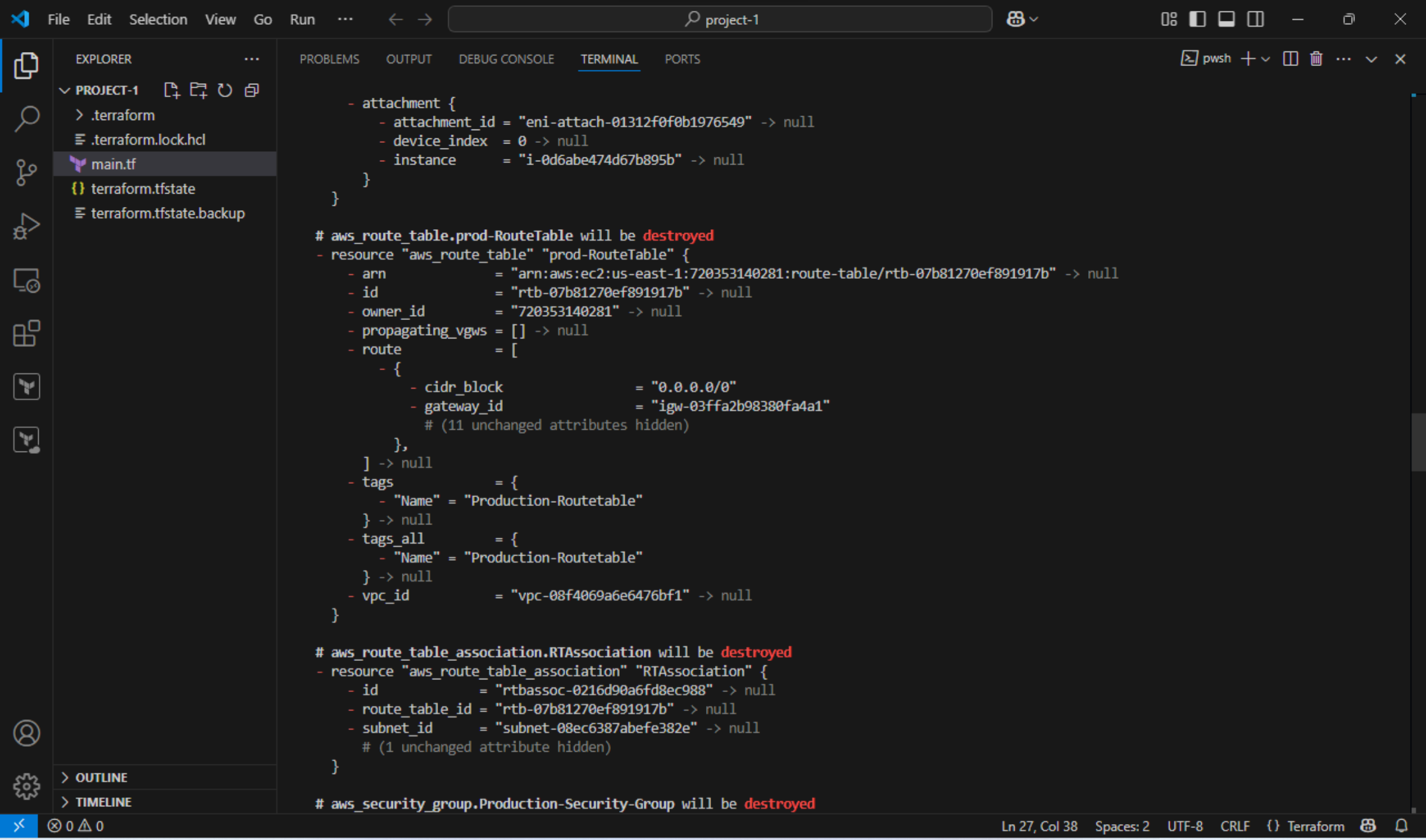

File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
- tags = {
- "Name" = "EIP-For-ENI"
} -> null
- tags_all = {
- "Name" = "EIP-For-ENI"
} -> null
- vpc = true -> null
# (4 unchanged attributes hidden)
}
# aws_eip_association.eip_attach_to_eni will be destroyed
- resource "aws_eip_association" "eip_attach_to_eni" {
- allocation_id = "eipalloc-019e1b1e3e2bd3686" -> null
- id = "eipassoc-0d82d32bca67ce728" -> null
- instance_id = "i-0d6abe474d67b895b" -> null
- network_interface_id = "eni-02f755443d35320a0" -> null
- private_ip_address = "10.0.1.50" -> null
- public_ip = "44.222.19.210" -> null
}
# aws_instance.webserver will be destroyed
- resource "aws_instance" "webserver" {
- ami = "ami-084568db4383264d4" -> null
- arn = "arn:aws:ec2:us-east-1:720353140281:instance/i-0d6abe474d67b895b" -> null
- associate_public_ip_address = true -> null
- availability_zone = "us-east-1a" -> null
- cpu_core_count = 1 -> null
- cpu_threads_per_core = 1 -> null
- disable_api_stop = false -> null
- disable_api_termination = false -> null
- ebs_optimized = false -> null
- get_password_data = false -> null
- hibernation = false -> null
- id = "i-0d6abe474d67b895b" -> null
- instance_initiated_shutdown_behavior = "stop" -> null
- instance_state = "running" -> null
- instance_type = "t2.micro" -> null
- ipv6_address_count = 0 -> null
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform

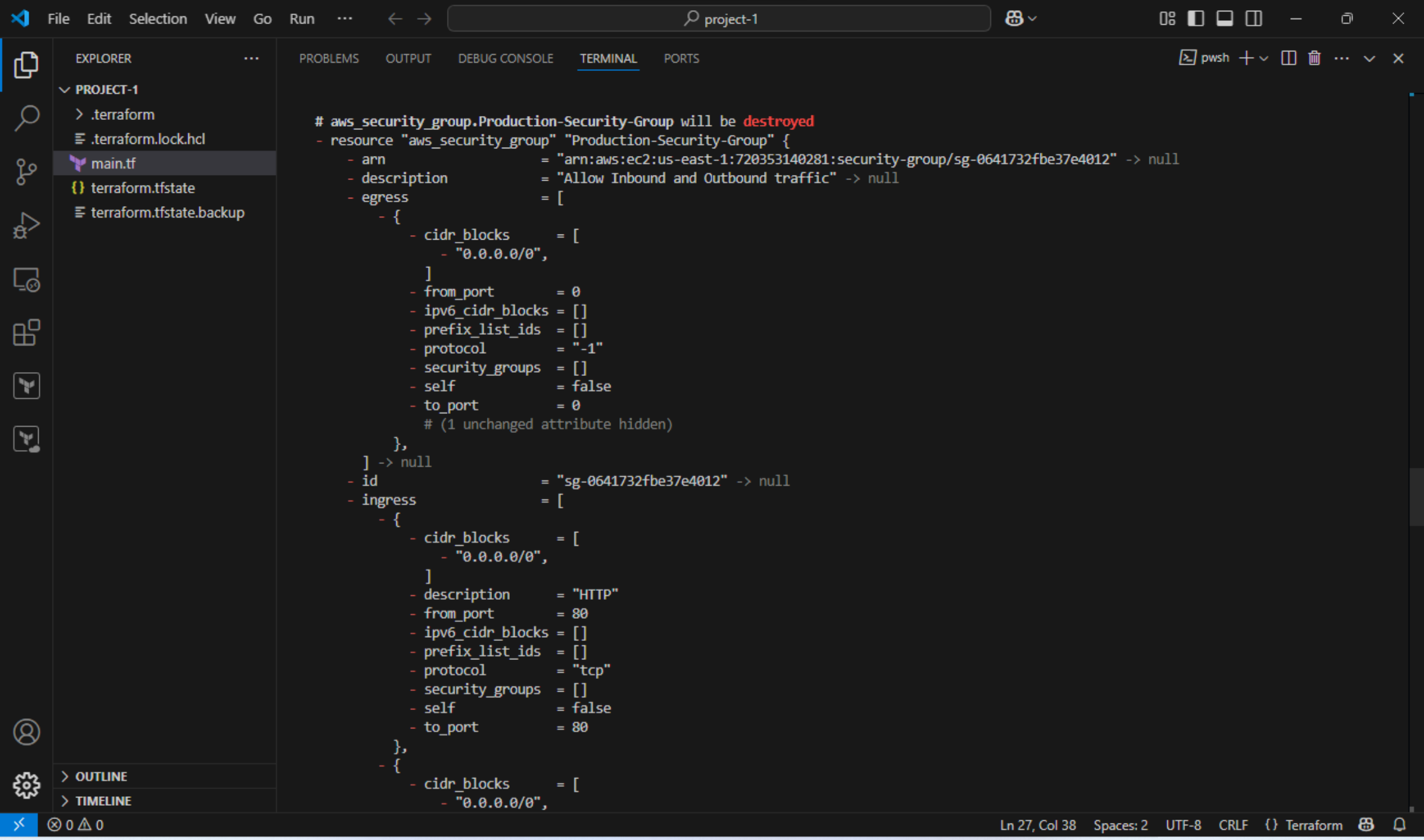

File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
- ipv6_address_count = 0 -> null
- ipv6_addresses = [] -> null
- key_name = "ec2.keypair" -> null
- monitoring = false -> null
- placement_partition_number = 0 -> null
- primary_network_interface_id = "eni-02f755443d35320a0" -> null
- private_dns = "ip-10-0-1-50.ec2.internal" -> null
- private_ip = "10.0.1.50" -> null
- public_ip = "44.222.19.210" -> null
- secondary_private_ips = [] -> null
- security_groups = [] -> null
- source_dest_check = true -> null
- subnet_id = "subnet-08ec6387abefe382e" -> null
- tags = {
- "Name" = "Ubuntu-Server"
} -> null
- tags_all = {
- "Name" = "Ubuntu-Server"
} -> null
- tenancy = "default" -> null
- user_data = "af45acea32513c59b78559271d36ba4e239d8314" -> null
- user_data_replace_on_change = false -> null
- vpc_security_group_ids = [
- "sg-0641732fbe37e4012",
] -> null
# (8 unchanged attributes hidden)
- capacity_reservation_specification {
- capacity_reservation_preference = "open" -> null
}
- cpu_options {
- core_count = 1 -> null
- threads_per_core = 1 -> null
# (1 unchanged attribute hidden)
}
- credit_specification {
⊗ 0 ⚠ 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF {} Terraform

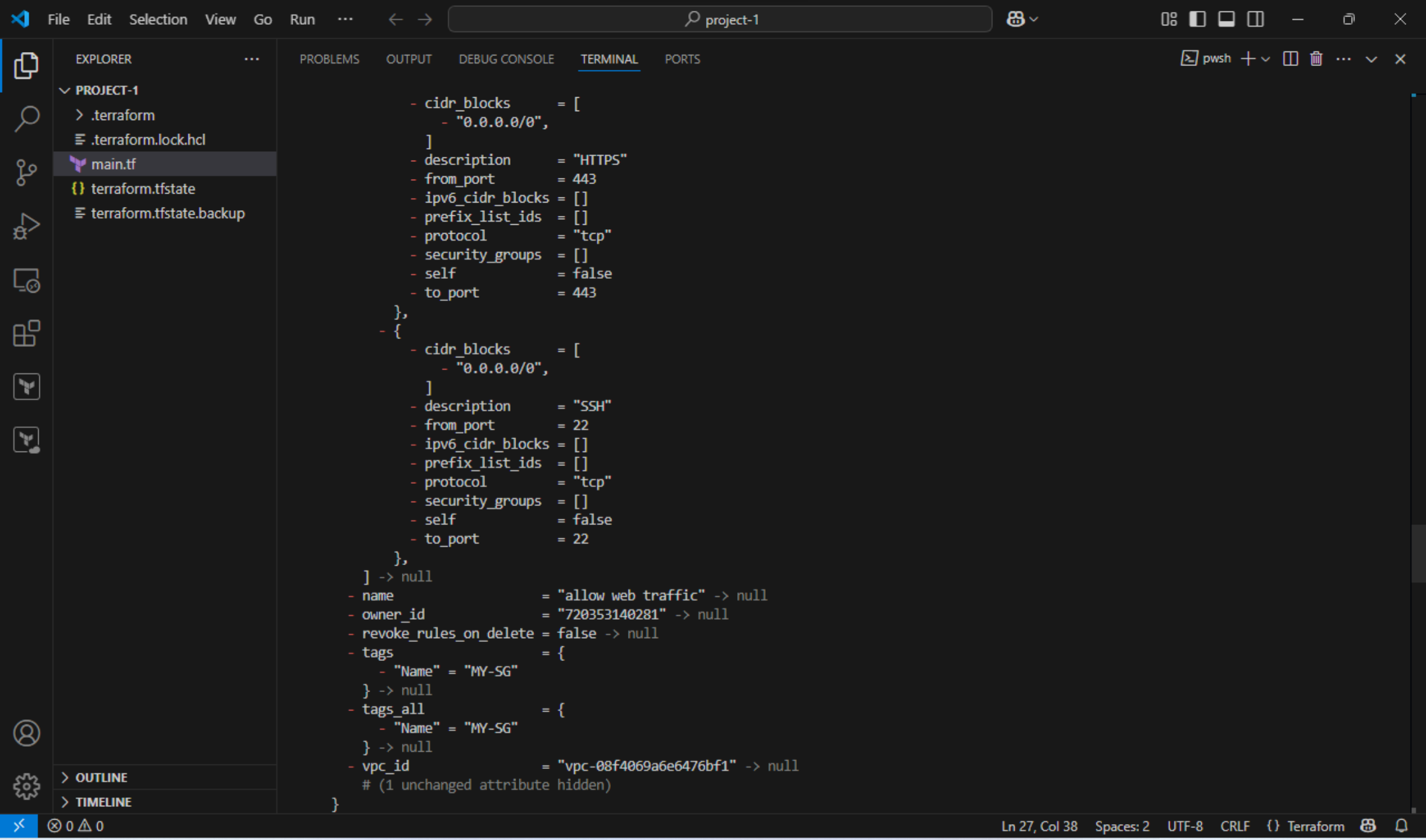

File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
- credit_specification {
- cpu_credits = "standard" -> null
}
- enclave_options {
- enabled = false -> null
}
- maintenance_options {
- auto_recovery = "default" -> null
}
- metadata_options {
- http_endpoint = "enabled" -> null
- http_protocol_ipv6 = "disabled" -> null
- http_put_response_hop_limit = 2 -> null
- http_tokens = "required" -> null
- instance_metadata_tags = "disabled" -> null
}
- network_interface {
- delete_on_termination = false -> null
- device_index = 0 -> null
- network_card_index = 0 -> null
- network_interface_id = "eni-02f755443d35320a0" -> null
}
- private_dns_name_options {
- enable_resource_name_dns_a_record = false -> null
- enable_resource_name_dns_aaaa_record = false -> null
- hostname_type = "ip-name" -> null
}
- root_block_device {
- delete_on_termination = true -> null
- device_name = "/dev/sda1" -> null
- encrypted = false -> null
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform

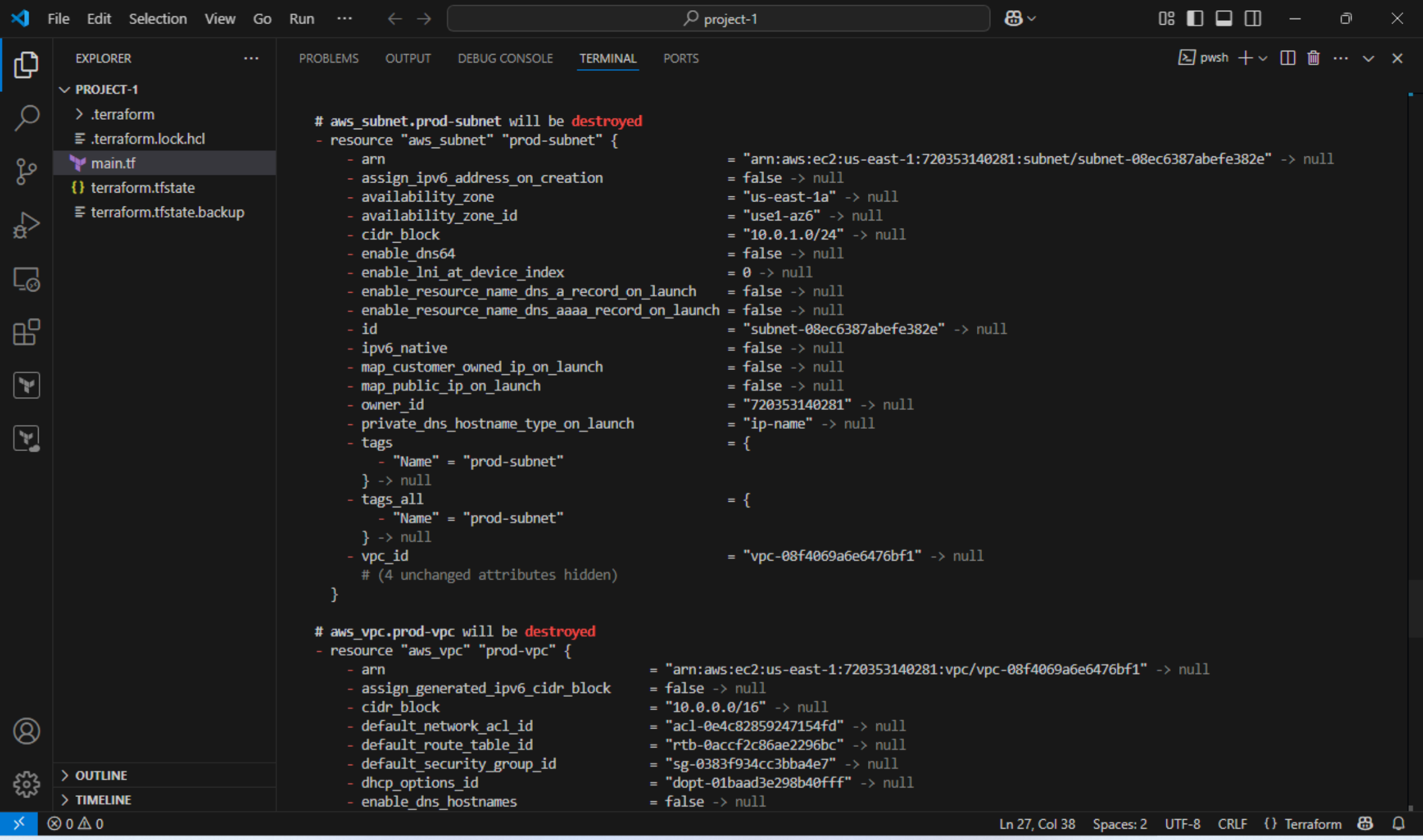

File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
- root_block_device {
- delete_on_termination = true -> null
- device_name = "/dev/sda1" -> null
- encrypted = false -> null
- iops = 3000 -> null
- tags = {} -> null
- tags_all = {} -> null
- throughput = 125 -> null
- volume_id = "vol-0c729ee3c18b23219" -> null
- volume_size = 8 -> null
- volume_type = "gp3" -> null
# (1 unchanged attribute hidden)
}
}
# aws_internet_gateway.InternetGateWay will be destroyed
- resource "aws_internet_gateway" "InternetGateWay" {
- arn = "arn:aws:ec2:us-east-1:720353140281:internet-gateway/igw-03ffa2b98380fa4a1" -> null
- id = "igw-03ffa2b98380fa4a1" -> null
- owner_id = "720353140281" -> null
- tags = {
- "Name" = "MY-IGW"
} -> null
- tags_all = {
- "Name" = "MY-IGW"
} -> null
- vpc_id = "vpc-08f4069a6e6476bf1" -> null
}
# aws_network_interface.NetworkInterface-webserver will be destroyed
- resource "aws_network_interface" "NetworkInterface-webserver" {
- arn = "arn:aws:ec2:us-east-1:720353140281:network-interface/eni-02f755443d35320a0" -> null
- id = "eni-02f755443d35320a0" -> null
- interface_type = "interface" -> null
- ipv4_prefix_count = 0 -> null
- ipv4_prefixes = [] -> null
- ipv6_address_count = 0 -> null
- ipv6_address_list = [] -> null
0 0
Ln 27, Col 38
Spaces: 2
UTF-8
CRLF
Terraform

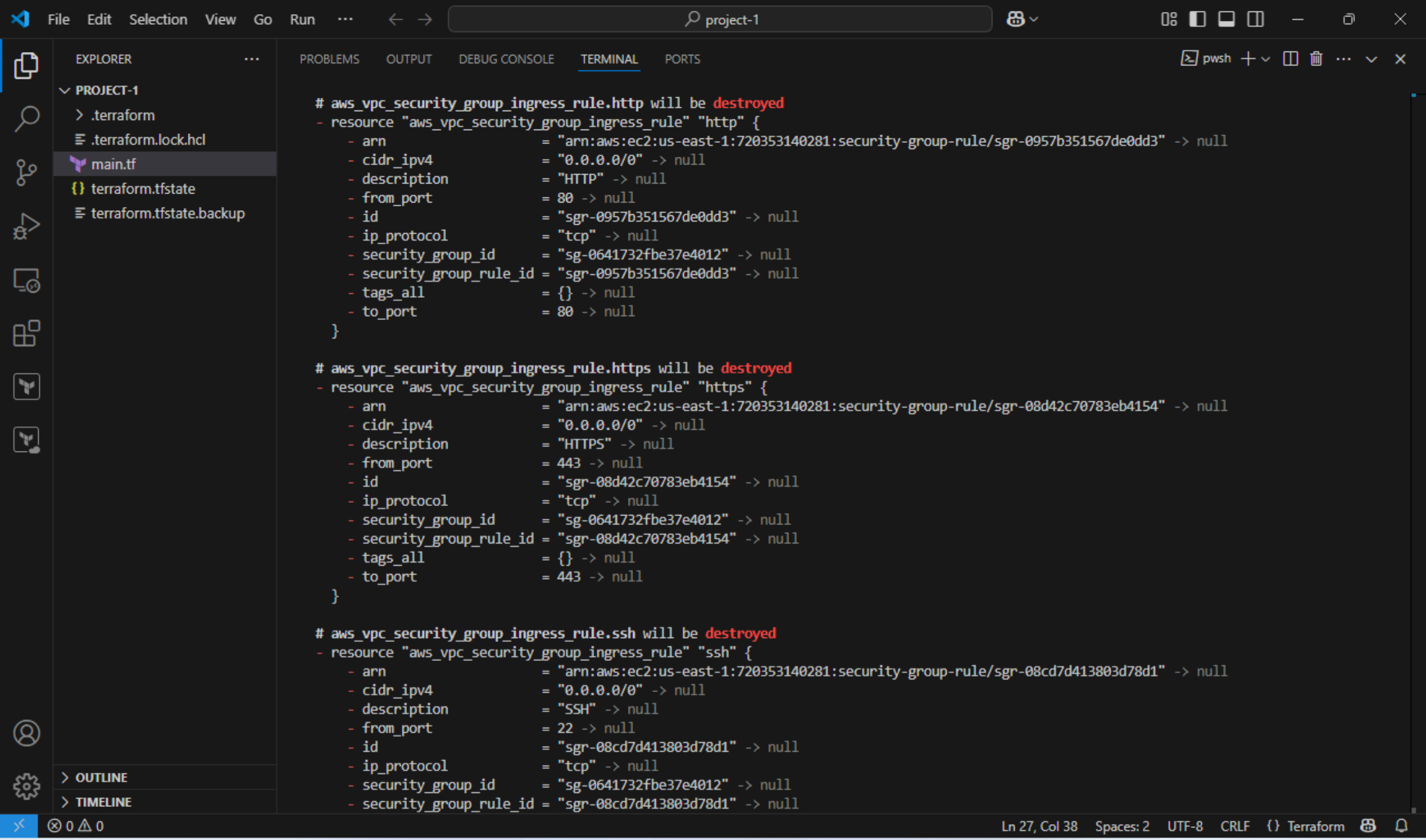

File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
# aws_network_interface.NetworkInterface-webserver will be destroyed
- resource "aws_network_interface" "NetworkInterface-webserver" {
- arn = "arn:aws:ec2:us-east-1:720353140281:network-interface/eni-02f755443d35320a0" -> null
- id = "eni-02f755443d35320a0" -> null
- interface_type = "interface" -> null
- ipv4_prefix_count = 0 -> null
- ipv4_prefixes = [] -> null
- ipv6_address_count = 0 -> null
- ipv6_address_list = [] -> null
- ipv6_address_list_enabled = false -> null
- ipv6_addresses = [] -> null
- ipv6_prefix_count = 0 -> null
- ipv6_prefixes = [] -> null
- mac_address = "0e:07:d7:af:7f:d5" -> null
- owner_id = "720353140281" -> null
- private_ip = "10.0.1.50" -> null
- private_ip_list = [
- "10.0.1.50",
] -> null
- private_ip_list_enabled = false -> null
- private_ips = [
- "10.0.1.50",
] -> null
- private_ips_count = 0 -> null
- security_groups = [
- "sg-0641732fbe37e4012",
] -> null
- source_dest_check = true -> null
- subnet_id = "subnet-08ec6387abefe382e" -> null
- tags = {
- "Name" = "My-ENI"
} -> null
- tags_all = {
- "Name" = "My-ENI"
} -> null
# (3 unchanged attributes hidden)
- attachment {
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform

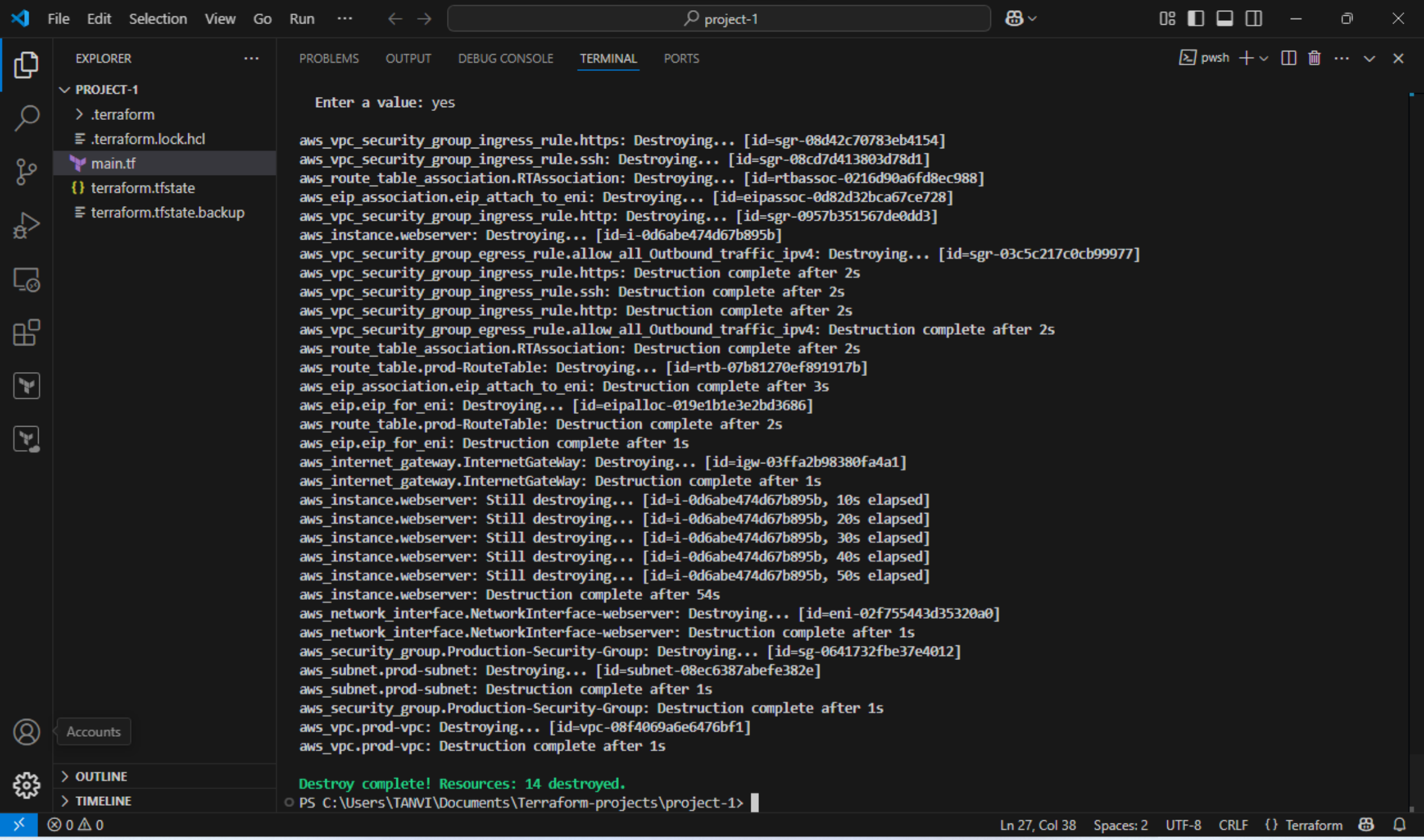

File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
- attachment {
- attachment_id = "eni-attach-01312f0f0b1976549" -> null
- device_index = 0 -> null
- instance = "i-0d6abe474d67b895b" -> null
}
}
# aws_route_table.prod-RouteTable will be destroyed
- resource "aws_route_table" "prod-RouteTable" {
- arn = "arn:aws:ec2:us-east-1:720353140281:route-table/rtb-07b81270ef891917b" -> null
- id = "rtb-07b81270ef891917b" -> null
- owner_id = "720353140281" -> null
- propagating_vgws = [] -> null
- route = [
- {
- cidr_block = "0.0.0.0/0"
- gateway_id = "igw-03ffa2b98380fa4a1"
(11 unchanged attributes hidden)
},
] -> null
- tags = {
- "Name" = "Production-Routetable"
} -> null
- tags_all = {
- "Name" = "Production-Routetable"
} -> null
- vpc_id = "vpc-08f4069a6e6476bf1" -> null
}
# aws_route_table_association.RTAssociation will be destroyed
- resource "aws_route_table_association" "RTAssociation" {
- id = "rtbassoc-0216d90a6fd8ec988" -> null
- route_table_id = "rtb-07b81270ef891917b" -> null
- subnet_id = "subnet-08ec6387abefe382e" -> null
# (1 unchanged attribute hidden)
}
# aws_security_group.Production-Security-Group will be destroyed
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform


File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
# aws_security_group.Production-Security-Group will be destroyed
- resource "aws_security_group" "Production-Security-Group" {
- arn = "arn:aws:ec2:us-east-1:720353140281:security-group/sg-0641732fbe37e4012" -> null
- description = "Allow Inbound and Outbound traffic" -> null
- egress = [
- {
- cidr_blocks = [
- "0.0.0.0/0",
]
- from_port = 0
- ipv6_cidr_blocks = []
- prefix_list_ids = []
- protocol = "-1"
- security_groups = []
- self = false
- to_port = 0
(1 unchanged attribute hidden)
},
] -> null
- id = "sg-0641732fbe37e4012" -> null
- ingress = [
- {
- cidr_blocks = [
- "0.0.0.0/0",
]
- description = "HTTP"
- from_port = 80
- ipv6_cidr_blocks = []
- prefix_list_ids = []
- protocol = "tcp"
- security_groups = []
- self = false
- to_port = 80
},
- {
- cidr_blocks = [
- "0.0.0.0/0",
0 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform


File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
- cidr_blocks = [
- "0.0.0.0/0",
]
- description = "HTTPS"
- from_port = 443
- ipv6_cidr_blocks = []
- prefix_list_ids = []
- protocol = "tcp"
- security_groups = []
- self = false
- to_port = 443
},
- {
- cidr_blocks = [
- "0.0.0.0/0",
]
- description = "SSH"
- from_port = 22
- ipv6_cidr_blocks = []
- prefix_list_ids = []
- protocol = "tcp"
- security_groups = []
- self = false
- to_port = 22
},
] -> null
- name = "allow web traffic" -> null
- owner_id = "720353140281" -> null
- revoke_rules_on_delete = false -> null
- tags = {
- "Name" = "MY-SG"
} -> null
- tags_all = {
- "Name" = "MY-SG"
} -> null
- vpc_id = "vpc-08f4069a6e6476bf1" -> null
# (1 unchanged attribute hidden)
}
0 0
Ln 27, Col 38
Spaces: 2
UTF-8
CRLF
Terraform


File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
# aws_subnet.prod-subnet will be destroyed
- resource "aws_subnet" "prod-subnet" {
- arn = "arn:aws:ec2:us-east-1:720353140281:subnet/subnet-08ec6387abefe382e" -> null
- assign_ipv6_address_on_creation = false -> null
- availability_zone = "us-east-1a" -> null
- availability_zone_id = "use1-az6" -> null
- cidr_block = "10.0.1.0/24" -> null
- enable_dns64 = false -> null
- enable_lni_at_device_index = 0 -> null
- enable_resource_name_dns_a_record_on_launch = false -> null
- enable_resource_name_dns_aaaa_record_on_launch = false -> null
- id = "subnet-08ec6387abefe382e" -> null
- ipv6_native = false -> null
- map_customer_owned_ip_on_launch = false -> null
- map_public_ip_on_launch = false -> null
- owner_id = "720353140281" -> null
- private_dns_hostname_type_on_launch = "ip-name" -> null
- tags = {
- "Name" = "prod-subnet"
} -> null
- tags_all = {
- "Name" = "prod-subnet"
} -> null
- vpc_id = "vpc-08f4069a6e6476bf1" -> null
# (4 unchanged attributes hidden)
}
# aws_vpc.prod-vpc will be destroyed
- resource "aws_vpc" "prod-vpc" {
- arn = "arn:aws:ec2:us-east-1:720353140281:vpc/vpc-08f4069a6e6476bf1" -> null
- assign_generated_ipv6_cidr_block = false -> null
- cidr_block = "10.0.0.0/16" -> null
- default_network_acl_id = "acl-0e4c82859247154fd" -> null
- default_route_table_id = "rtb-0accf2c86ae2296bc" -> null
- default_security_group_id = "sg-0383f934cc3bba4e7" -> null
- dhcp_options_id = "dopt-01baad3e298b40fff" -> null
- enable_dns_hostnames = false -> null
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF {} Terraform


File Edit Selection View Go Run ···
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
# aws_vpc_security_group_ingress_rule.http will be destroyed
- resource "aws_vpc_security_group_ingress_rule" "http" {
- arn = "arn:aws:ec2:us-east-1:720353140281:security-group-rule/sgr-0957b351567de0dd3" -> null
- cidr_ipv4 = "0.0.0.0/0" -> null
- description = "HTTP" -> null
- from_port = 80 -> null
- id = "sgr-0957b351567de0dd3" -> null
- ip_protocol = "tcp" -> null
- security_group_id = "sg-0641732fbe37e4012" -> null
- security_group_rule_id = "sgr-0957b351567de0dd3" -> null
- tags_all = {} -> null
- to_port = 80 -> null
}
# aws_vpc_security_group_ingress_rule.https will be destroyed
- resource "aws_vpc_security_group_ingress_rule" "https" {
- arn = "arn:aws:ec2:us-east-1:720353140281:security-group-rule/sgr-08d42c70783eb4154" -> null
- cidr_ipv4 = "0.0.0.0/0" -> null
- description = "HTTPS" -> null
- from_port = 443 -> null
- id = "sgr-08d42c70783eb4154" -> null
- ip_protocol = "tcp" -> null
- security_group_id = "sg-0641732fbe37e4012" -> null
- security_group_rule_id = "sgr-08d42c70783eb4154" -> null
- tags_all = {} -> null
- to_port = 443 -> null
}
# aws_vpc_security_group_ingress_rule.ssh will be destroyed
- resource "aws_vpc_security_group_ingress_rule" "ssh" {
- arn = "arn:aws:ec2:us-east-1:720353140281:security-group-rule/sgr-08cd7d413803d78d1" -> null
- cidr_ipv4 = "0.0.0.0/0" -> null
- description = "SSH" -> null
- from_port = 22 -> null
- id = "sgr-08cd7d413803d78d1" -> null
- ip_protocol = "tcp" -> null
- security_group_id = "sg-0641732fbe37e4012" -> null
- security_group_rule_id = "sgr-08cd7d413803d78d1" -> null
⊗ 0 ⚠ 0
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF {} Terraform


File Edit Selection View Go Run
project-1
EXPLORER
PROJECT-1
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh
Enter a value: yes
aws_vpc_security_group_ingress_rule.https: Destroying... [id=sgr-08d42c70783eb4154]
aws_vpc_security_group_ingress_rule.ssh: Destroying... [id=sgr-08cd7d413803d78d1]
aws_route_table_association.RTAssociation: Destroying... [id=rtbassoc-0216d90a6fd8ec988]
aws_eip_association.eip_attach_to_eni: Destroying... [id=eipassoc-0d82d32bca67ce728]
aws_vpc_security_group_ingress_rule.http: Destroying... [id=sgr-0957b351567de0dd3]
aws_instance.webserver: Destroying... [id=i-0d6abe474d67b895b]
aws_vpc_security_group_egress_rule.allow_all_Outbound_traffic_ipv4: Destroying... [id=sgr-03c5c217c0cb99977]
aws_vpc_security_group_ingress_rule.https: Destruction complete after 2s
aws_vpc_security_group_ingress_rule.ssh: Destruction complete after 2s
aws_vpc_security_group_ingress_rule.http: Destruction complete after 2s
aws_vpc_security_group_egress_rule.allow_all_Outbound_traffic_ipv4: Destruction complete after 2s
aws_route_table_association.RTAssociation: Destruction complete after 2s
aws_route_table.prod-RouteTable: Destroying... [id=rtb-07b81270ef891917b]
aws_eip_association.eip_attach_to_eni: Destruction complete after 3s
aws_eip.eip_for_eni: Destroying... [id=eipalloc-019e1b1e3e2bd3686]
aws_route_table.prod-RouteTable: Destruction complete after 2s
aws_eip.eip_for_eni: Destruction complete after 1s
aws_internet_gateway.InternetGateWay: Destroying... [id=igw-03ffa2b98380fa4a1]
aws_internet_gateway.InternetGateWay: Destruction complete after 1s
aws_instance.webserver: Still destroying... [id=i-0d6abe474d67b895b, 10s elapsed]
aws_instance.webserver: Still destroying... [id=i-0d6abe474d67b895b, 20s elapsed]
aws_instance.webserver: Still destroying... [id=i-0d6abe474d67b895b, 30s elapsed]
aws_instance.webserver: Still destroying... [id=i-0d6abe474d67b895b, 40s elapsed]
aws_instance.webserver: Still destroying... [id=i-0d6abe474d67b895b, 50s elapsed]
aws_instance.webserver: Destruction complete after 54s
aws_network_interface.NetworkInterface-webserver: Destroying... [id=eni-02f755443d35320a0]
aws_network_interface.NetworkInterface-webserver: Destruction complete after 1s
aws_security_group.Production-Security-Group: Destroying... [id=sg-0641732fbe37e4012]
aws_subnet.prod-subnet: Destroying... [id=subnet-08ec6387abefe382e]
aws_subnet.prod-subnet: Destruction complete after 1s
aws_security_group.Production-Security-Group: Destruction complete after 1s
aws_vpc.prod-vpc: Destroying... [id=vpc-08f4069a6e6476bf1]
aws_vpc.prod-vpc: Destruction complete after 1s
Destroy complete! Resources: 14 destroyed.
PS C:\Users\TANVI\Documents\Terraform-projects\project-1>
Accounts
OUTLINE
TIMELINE
Ln 27, Col 38 Spaces: 2 UTF-8 CRLF Terraform