Fatima Jinnah Women University

Subject: Cloud Computing

Lab 11

Name:

Tehreem khan(5-B)

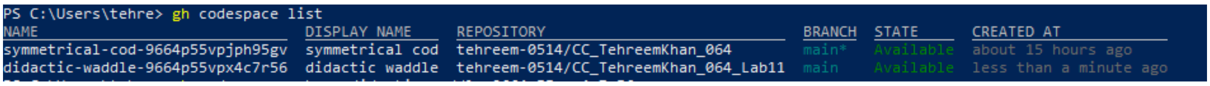
Registration number:

2023-BSE-064

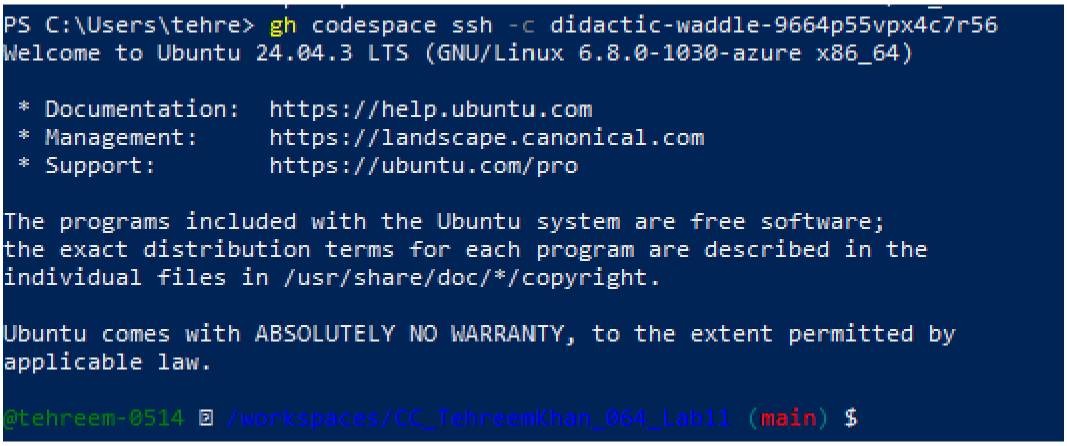
Submitted To:

Sir Shoaib

Task 0:

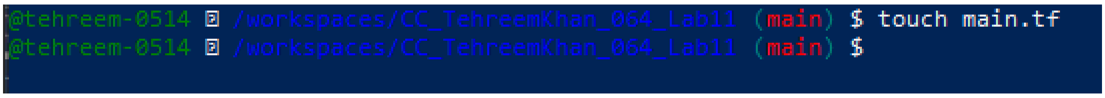


```
PS C:\Users\tehre> gh codespace list
NAME                               DISPLAY NAME        REPOSITORY                              BRANCH   STATE       CREATED AT
symmetrical-cod-9664p55vpjph95gv   symmetrical cod     tehreem-0514/CC_TehreemKhan_064         main*    Available   about 15 hours ago
didactic-waddle-9664p55vpx4c7r56   didactic waddle     tehreem-0514/CC_TehreemKhan_064_Lab11   main     Available   less than a minute ago
```

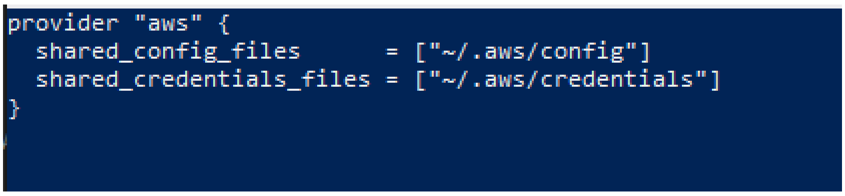


```
PS C:\Users\tehre> gh codespace ssh -c didactic-waddle-9664p55vpx4c7r56
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-1030-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

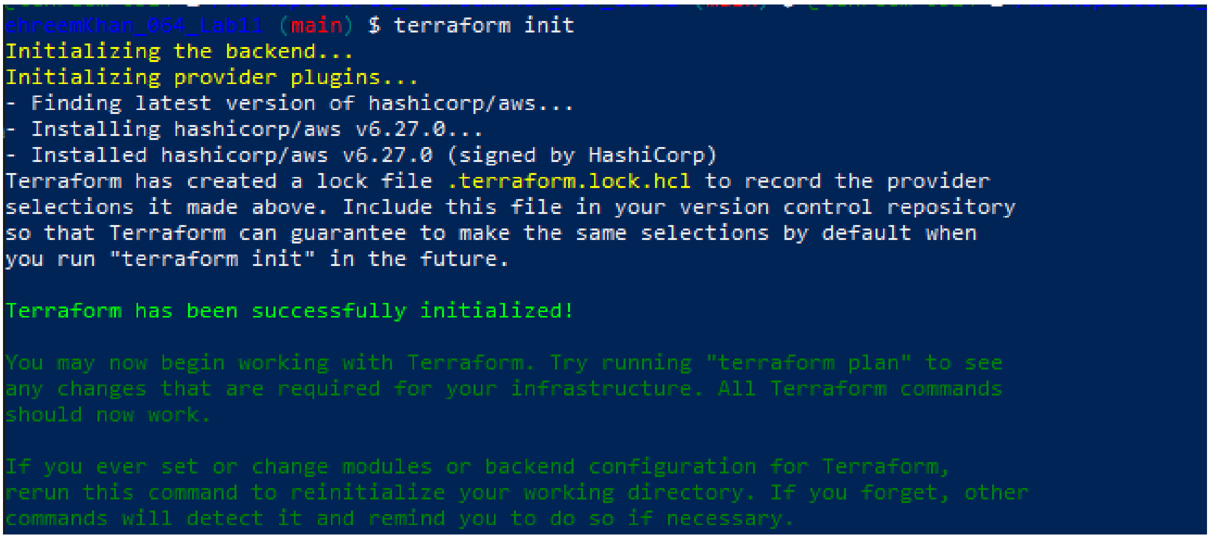
Task 1:



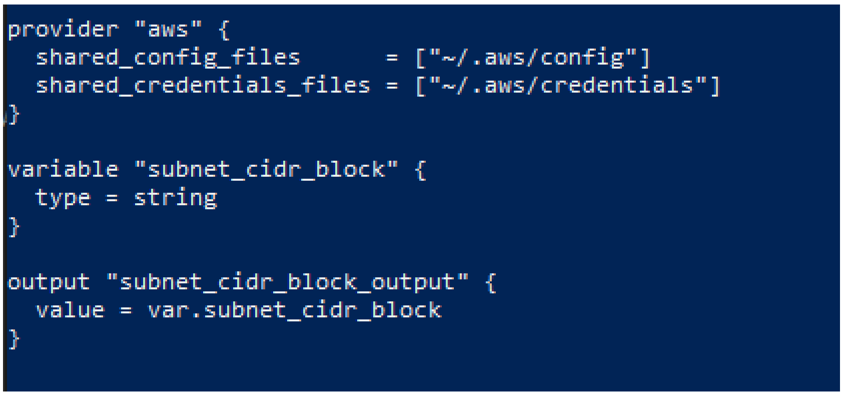
```
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ touch main.tf
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



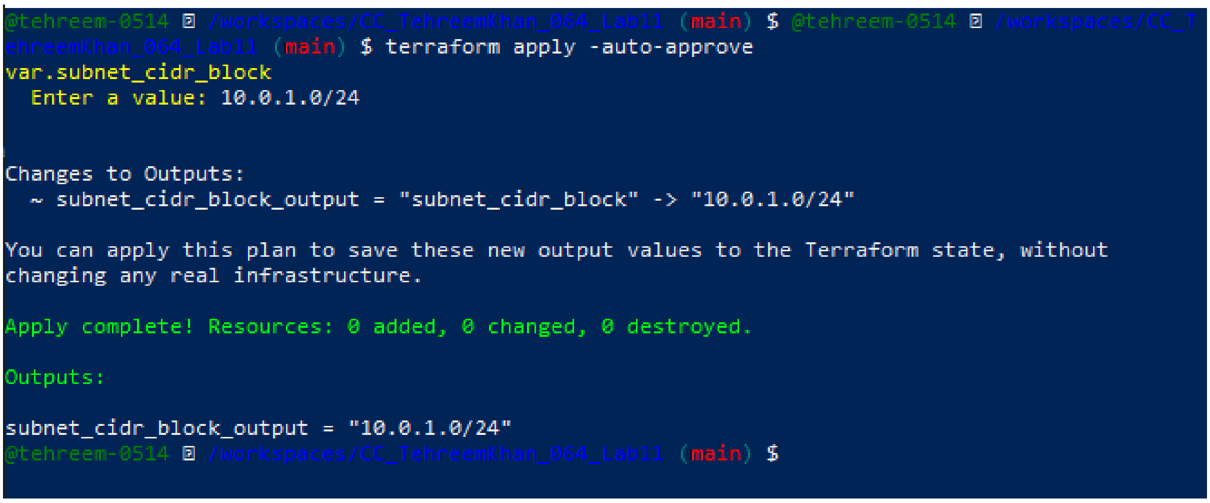
```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}
```



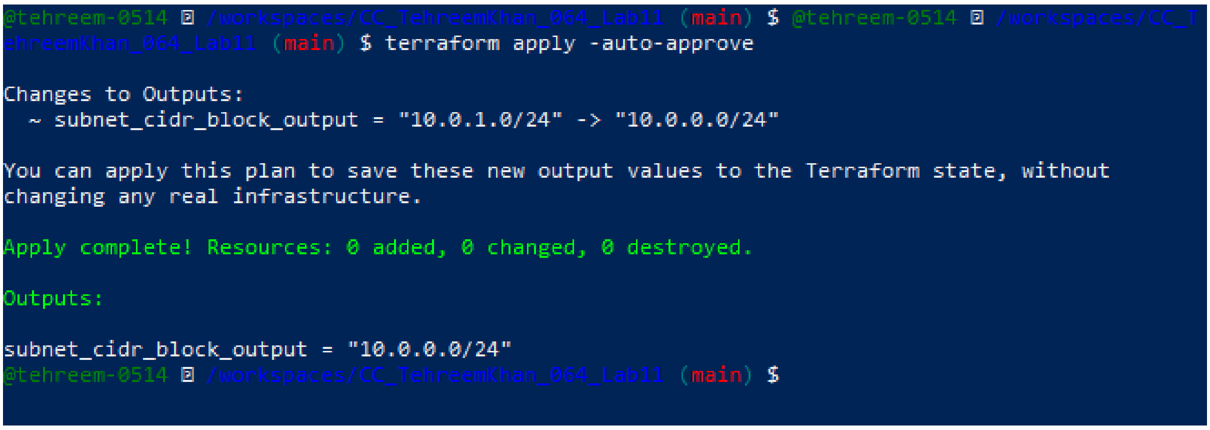
```
ehreemKhan_064_Lab11 (main) $ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v6.27.0...
- Installed hashicorp/aws v6.27.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl 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.
```



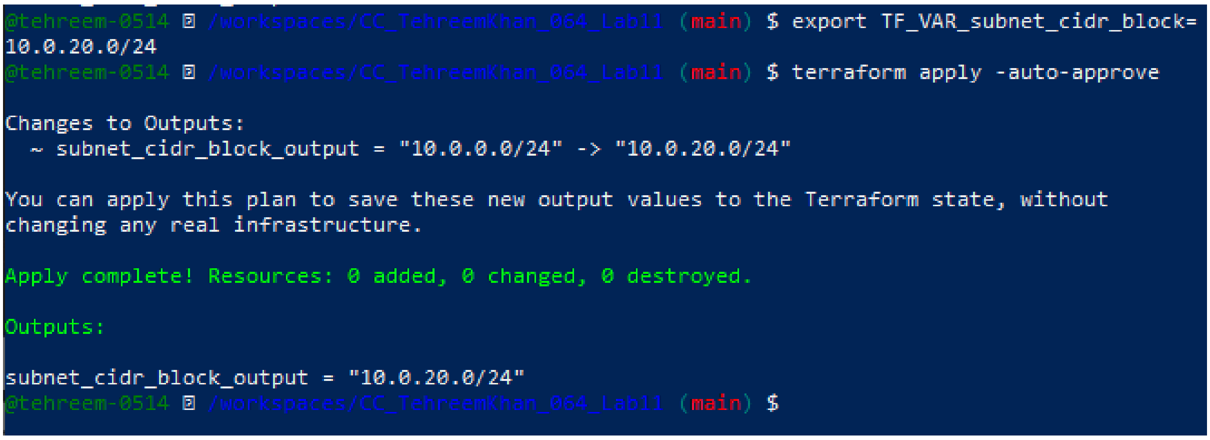
```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

variable "subnet_cidr_block" {
  type = string
}

output "subnet_cidr_block_output" {
  value = var.subnet_cidr_block
}
```



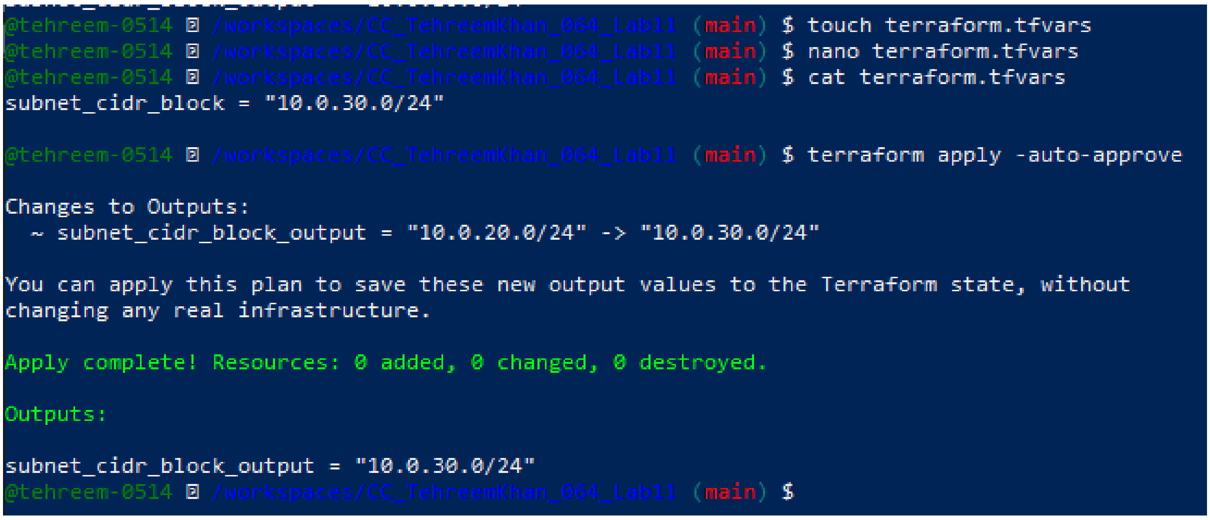
```
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ @tehreem-0514 ⍰ /workspaces/CC_T
ehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve
var.subnet_cidr_block
  Enter a value: 10.0.1.0/24


Changes to Outputs:
  ~ subnet_cidr_block_output = "subnet_cidr_block" -> "10.0.1.0/24"

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

subnet_cidr_block_output = "10.0.1.0/24"
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



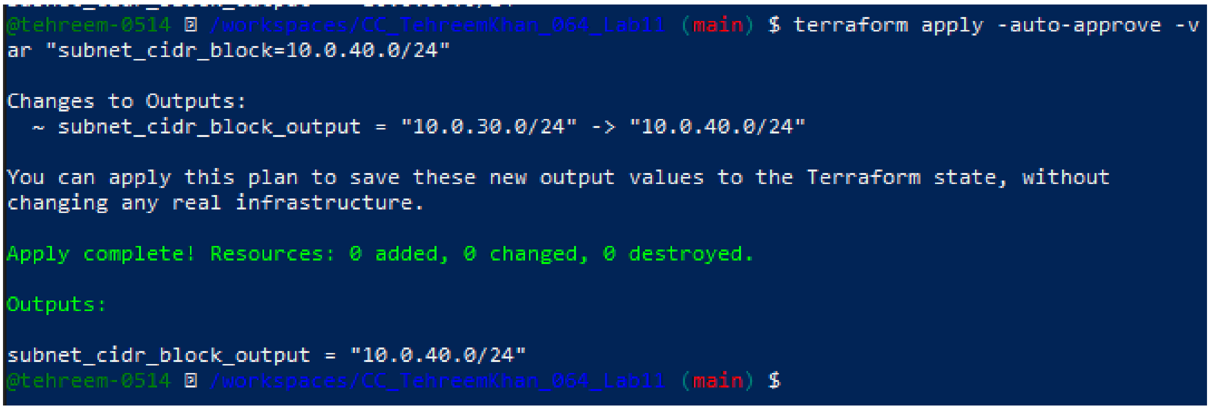
```
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ @tehreem-0514 ⍰ /workspaces/CC_T
ehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  ~ subnet_cidr_block_output = "10.0.1.0/24" -> "10.0.0.0/24"

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

subnet_cidr_block_output = "10.0.0.0/24"
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



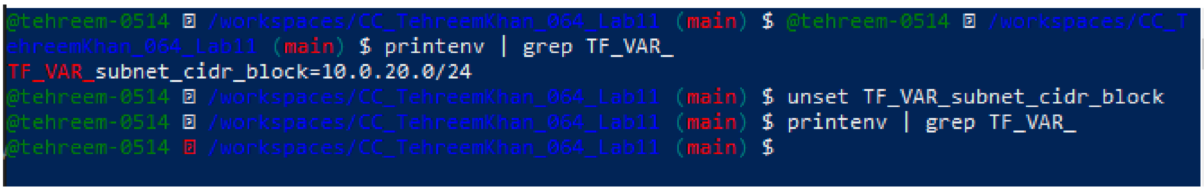
```
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ export TF_VAR_subnet_cidr_block=
10.0.20.0/24
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  ~ subnet_cidr_block_output = "10.0.0.0/24" -> "10.0.20.0/24"

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

subnet_cidr_block_output = "10.0.20.0/24"
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



```
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ touch terraform.tfvars
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano terraform.tfvars
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ cat terraform.tfvars
subnet_cidr_block = "10.0.30.0/24"

@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  ~ subnet_cidr_block_output = "10.0.20.0/24" -> "10.0.30.0/24"

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

subnet_cidr_block_output = "10.0.30.0/24"
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

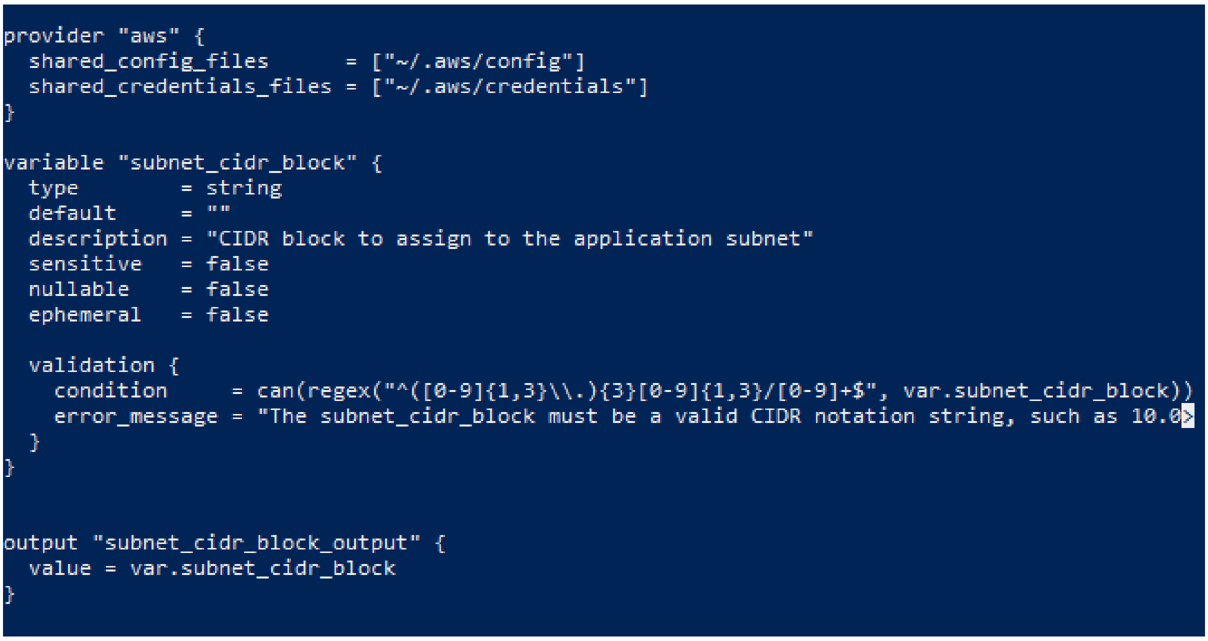


```
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve -v
ar "subnet_cidr_block=10.0.40.0/24"

Changes to Outputs:
  ~ subnet_cidr_block_output = "10.0.30.0/24" -> "10.0.40.0/24"

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

subnet_cidr_block_output = "10.0.40.0/24"
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

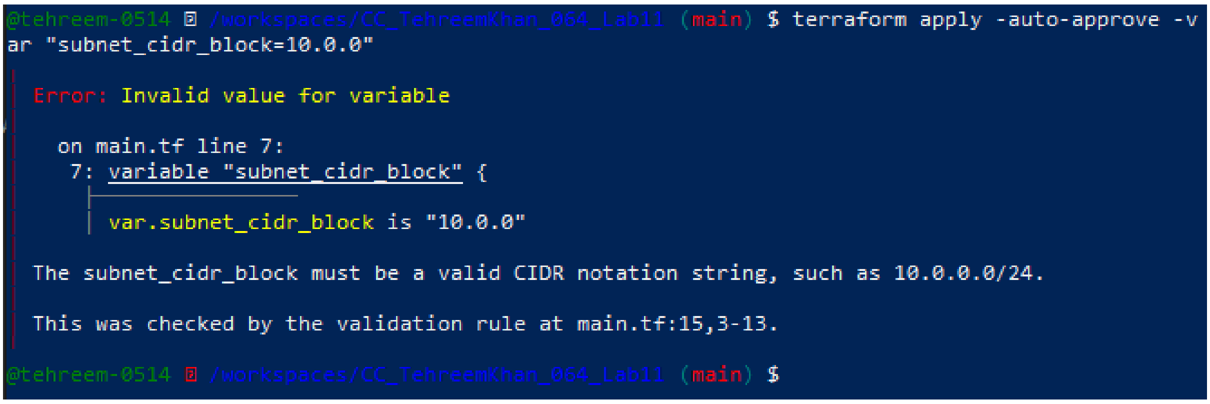


```
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ @tehreem-0514 ▯ /workspaces/CC_T
ehreemKhan_064_Lab11 (main) $ printenv | grep TF_VAR_
TF_VAR_subnet_cidr_block=10.0.20.0/24
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ unset TF_VAR_subnet_cidr_block
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ printenv | grep TF_VAR_
@tehreem-0514 ▯ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

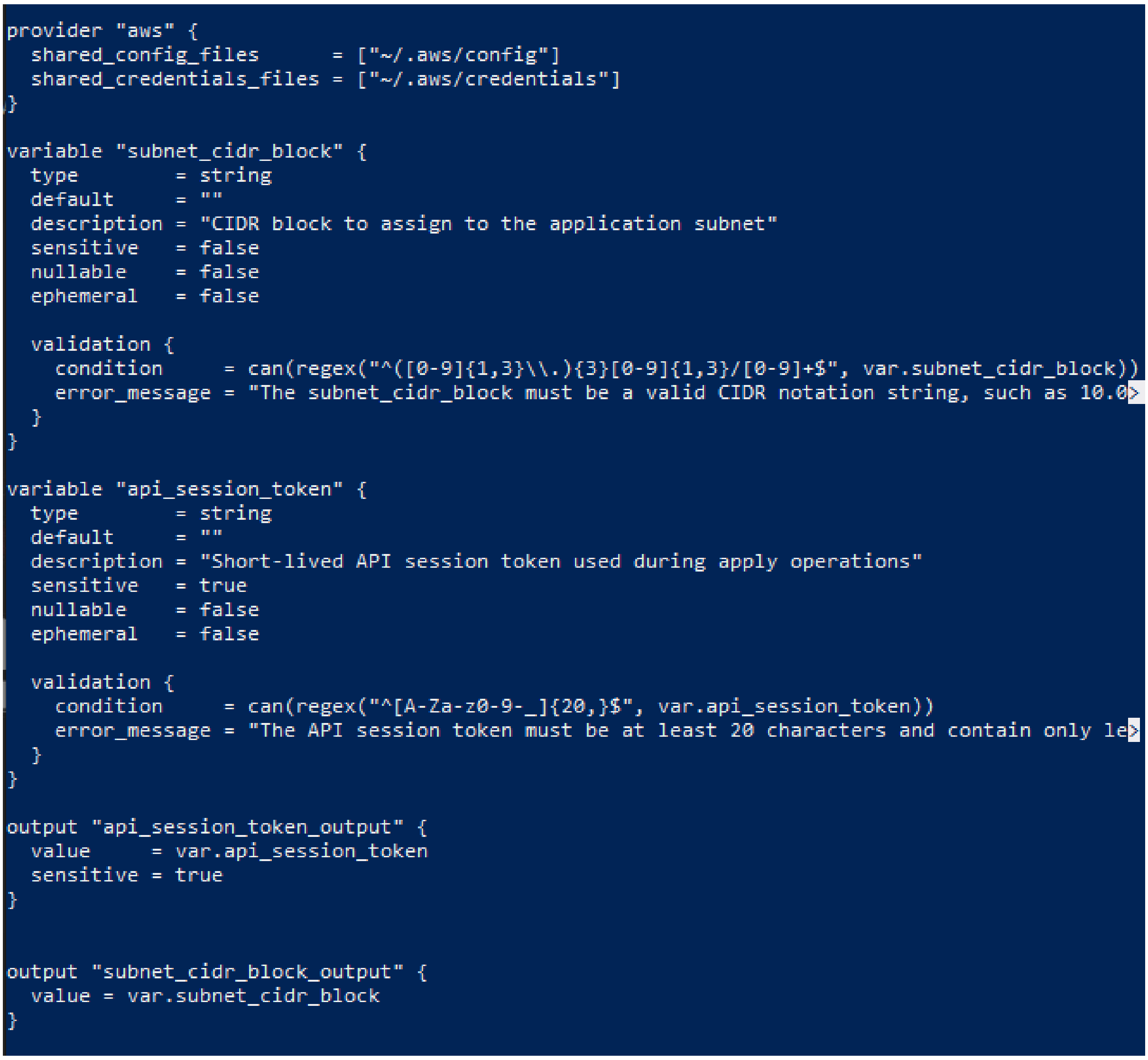
Task 2:



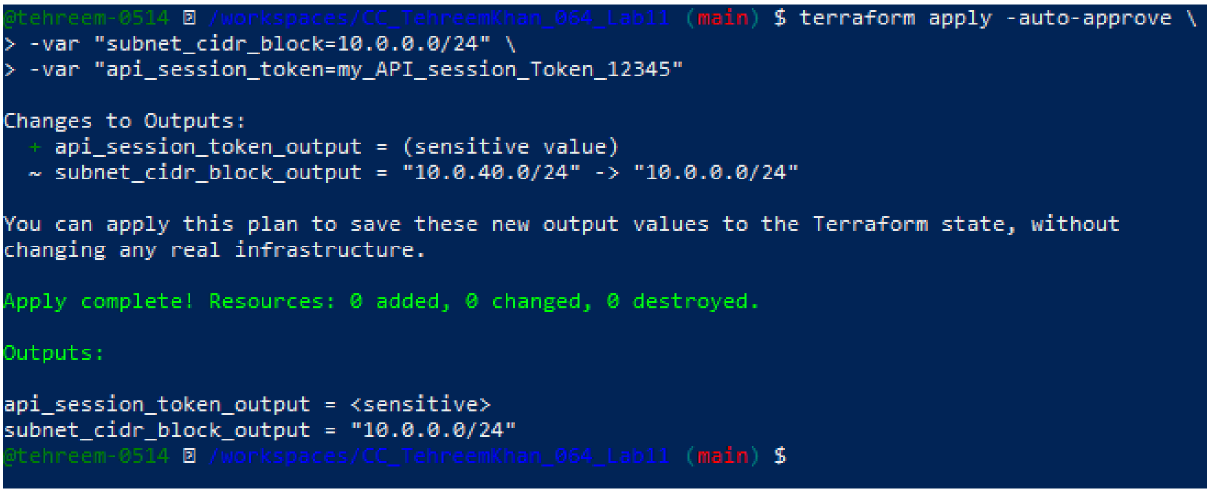
```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

variable "subnet_cidr_block" {
  type        = string
  default     = ""
  description = "CIDR block to assign to the application subnet"
  sensitive   = false
  nullable    = false
  ephemeral   = false

  validation {
    condition     = can(regex("^([0-9]{1,3}\\.){3}[0-9]{1,3}/[0-9]+$", var.subnet_cidr_block))
    error_message = "The subnet_cidr_block must be a valid CIDR notation string, such as 10.0>
  }
}


output "subnet_cidr_block_output" {
  value = var.subnet_cidr_block
}
```



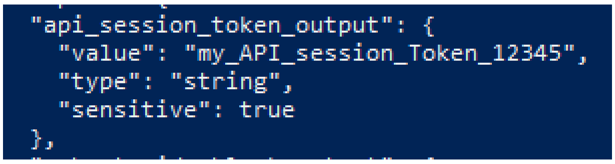
```
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve -v
ar "subnet_cidr_block=10.0.0"
│
│ Error: Invalid value for variable
│
│   on main.tf line 7:
│    7: variable "subnet_cidr_block" {
│      ├────────────────
│      │ var.subnet_cidr_block is "10.0.0"
│
│ The subnet_cidr_block must be a valid CIDR notation string, such as 10.0.0.0/24.
│
│ This was checked by the validation rule at main.tf:15,3-13.
╵
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



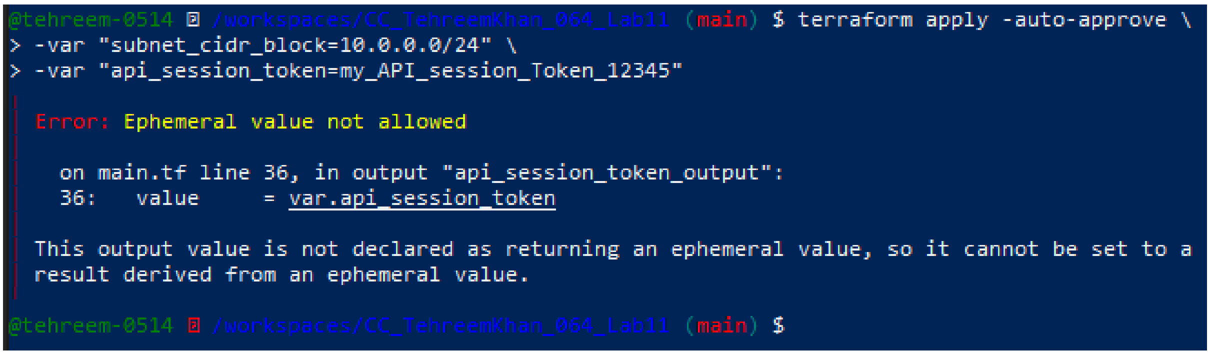
```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

variable "subnet_cidr_block" {
  type        = string
  default     = ""
  description = "CIDR block to assign to the application subnet"
  sensitive   = false
  nullable    = false
  ephemeral   = false

  validation {
    condition     = can(regex("^([0-9]{1,3}\\.){3}[0-9]{1,3}/[0-9]+$", var.subnet_cidr_block))
    error_message = "The subnet_cidr_block must be a valid CIDR notation string, such as 10.0>
  }
}

variable "api_session_token" {
  type        = string
  default     = ""
  description = "Short-lived API session token used during apply operations"
  sensitive   = true
  nullable    = false
  ephemeral   = false

  validation {
    condition     = can(regex("^[A-Za-z0-9-_]{20,}$", var.api_session_token))
    error_message = "The API session token must be at least 20 characters and contain only le>
  }
}

output "api_session_token_output" {
  value     = var.api_session_token
  sensitive = true
}


output "subnet_cidr_block_output" {
  value = var.subnet_cidr_block
}
```



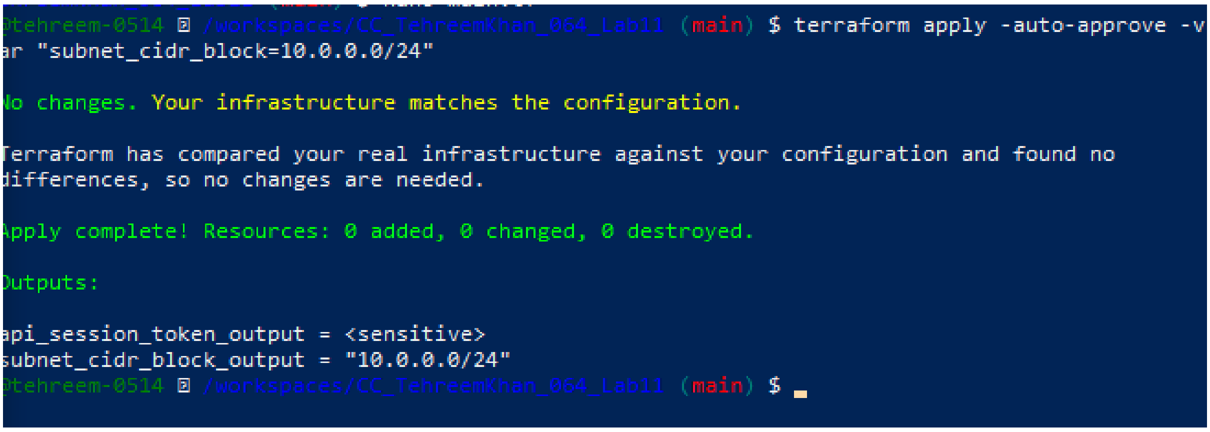
```
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve \
> -var "subnet_cidr_block=10.0.0.0/24" \
> -var "api_session_token=my_API_session_Token_12345"

Changes to Outputs:
  + api_session_token_output = (sensitive value)
  ~ subnet_cidr_block_output = "10.0.40.0/24" -> "10.0.0.0/24"

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

api_session_token_output = <sensitive>
subnet_cidr_block_output = "10.0.0.0/24"
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



```
  "api_session_token_output": {
    "value": "my_API_session_Token_12345",
    "type": "string",
    "sensitive": true
  },
```

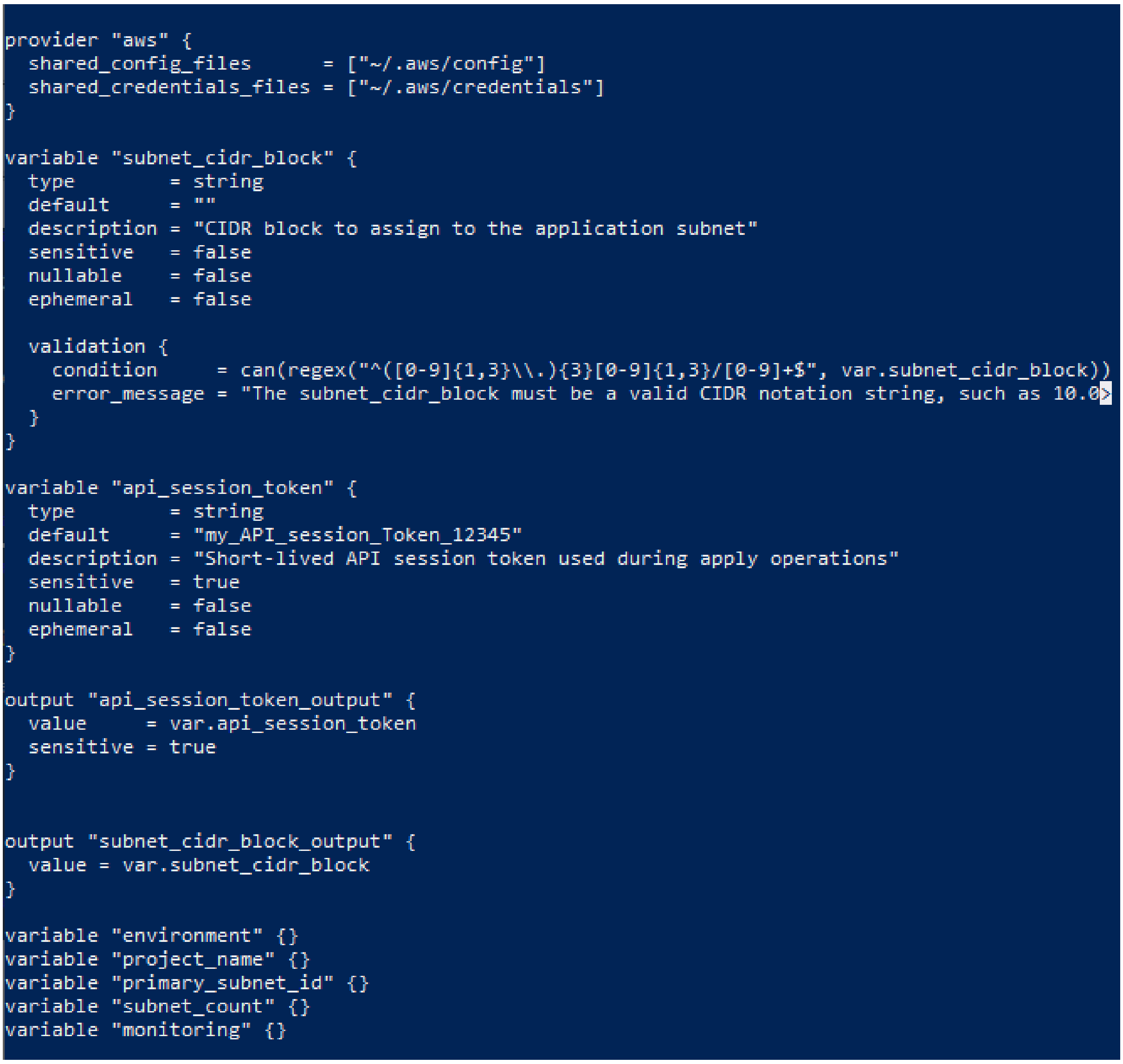


```
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve \
> -var "subnet_cidr_block=10.0.0.0/24" \
> -var "api_session_token=my_API_session_Token_12345"

│ Error: Ephemeral value not allowed
│
│   on main.tf line 36, in output "api_session_token_output":
│   36:   value      = var.api_session_token
│
│ This output value is not declared as returning an ephemeral value, so it cannot be set to a
│ result derived from an ephemeral value.

@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

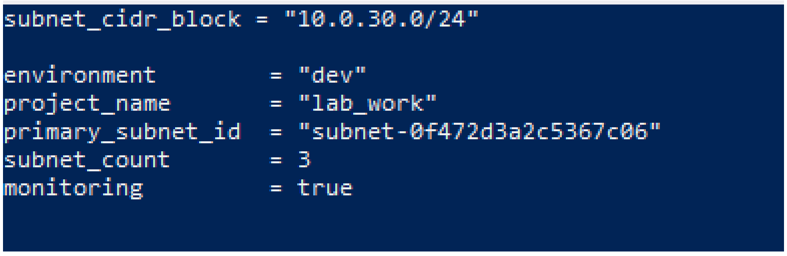


```
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve -v
ar "subnet_cidr_block=10.0.0.0/24"

No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no
differences, so no changes are needed.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

api_session_token_output = <sensitive>
subnet_cidr_block_output = "10.0.0.0/24"
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

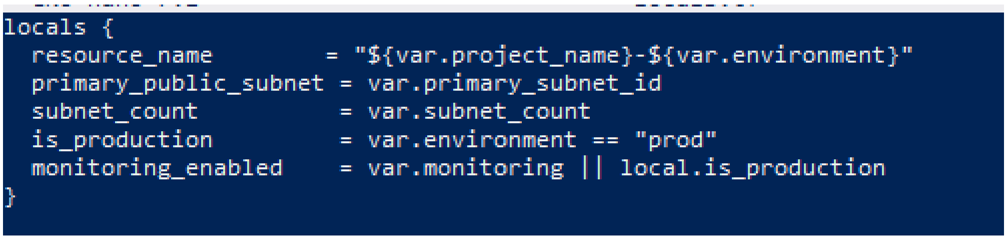
Task 3:



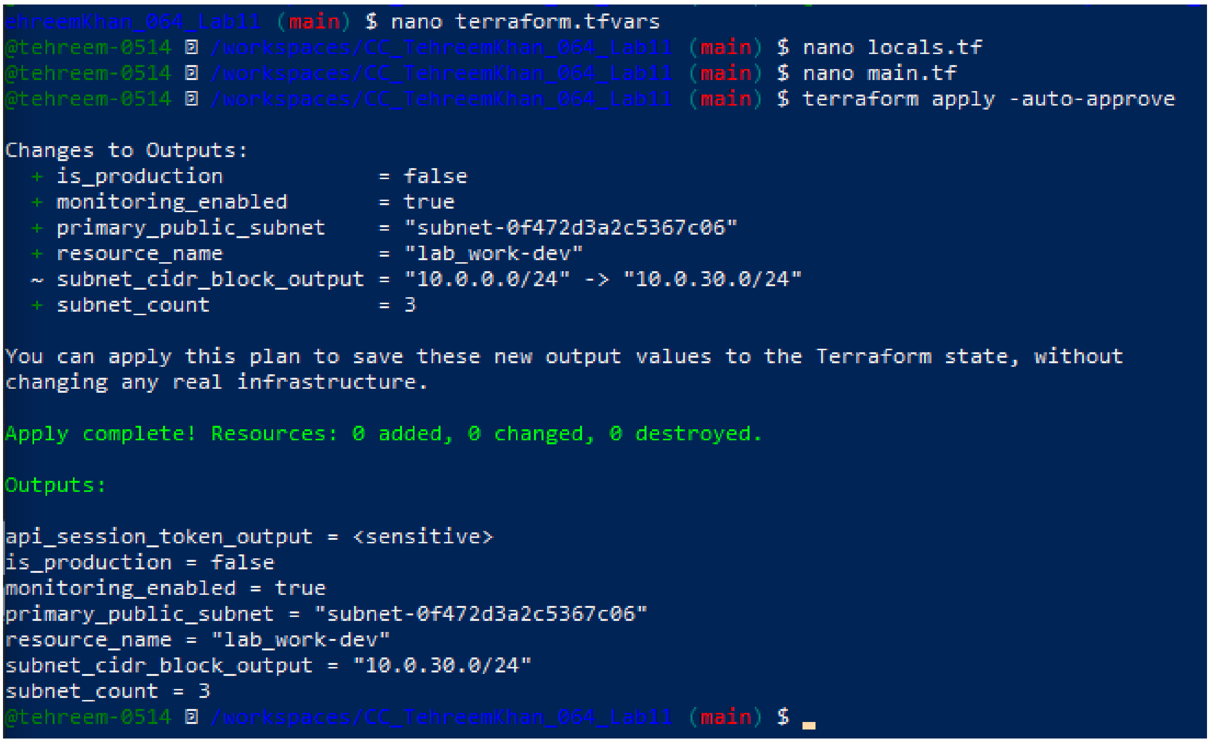
```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

variable "subnet_cidr_block" {
  type        = string
  default     = ""
  description = "CIDR block to assign to the application subnet"
  sensitive   = false
  nullable    = false
  ephemeral   = false

  validation {
    condition     = can(regex("^([0-9]{1,3}\\.){3}[0-9]{1,3}/[0-9]+$", var.subnet_cidr_block))
    error_message = "The subnet_cidr_block must be a valid CIDR notation string, such as 10.0>
  }
}

variable "api_session_token" {
  type        = string
  default     = "my_API_session_Token_12345"
  description = "Short-lived API session token used during apply operations"
  sensitive   = true
  nullable    = false
  ephemeral   = false
}

output "api_session_token_output" {
  value     = var.api_session_token
  sensitive = true
}


output "subnet_cidr_block_output" {
  value = var.subnet_cidr_block
}

variable "environment" {}
variable "project_name" {}
variable "primary_subnet_id" {}
variable "subnet_count" {}
variable "monitoring" {}
```



```
subnet_cidr_block = "10.0.30.0/24"

environment        = "dev"
project_name       = "lab_work"
primary_subnet_id  = "subnet-0f472d3a2c5367c06"
subnet_count       = 3
monitoring         = true
```

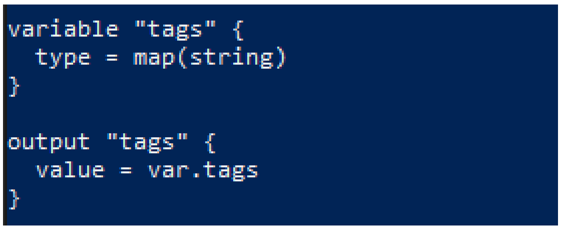


```
locals {
  resource_name         = "${var.project_name}-${var.environment}"
  primary_public_subnet = var.primary_subnet_id
  subnet_count          = var.subnet_count
  is_production         = var.environment == "prod"
  monitoring_enabled    = var.monitoring || local.is_production
}
```

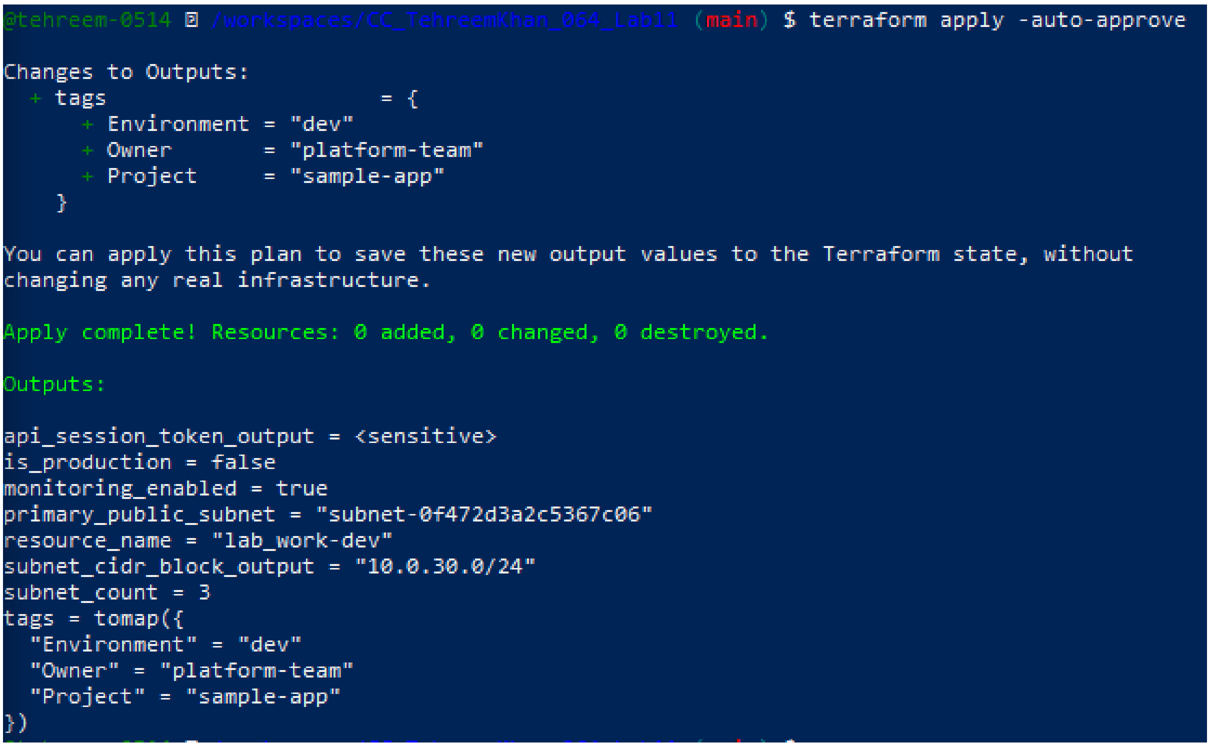


```
ehreemKhan_064_Lab11 (main) $ nano terraform.tfvars
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano locals.tf
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano main.tf
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  + is_production            = false
  + monitoring_enabled       = true
  + primary_public_subnet    = "subnet-0f472d3a2c5367c06"
  + resource_name            = "lab_work-dev"
  ~ subnet_cidr_block_output = "10.0.0.0/24" -> "10.0.30.0/24"
  + subnet_count             = 3

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

api_session_token_output = <sensitive>
is_production = false
monitoring_enabled = true
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

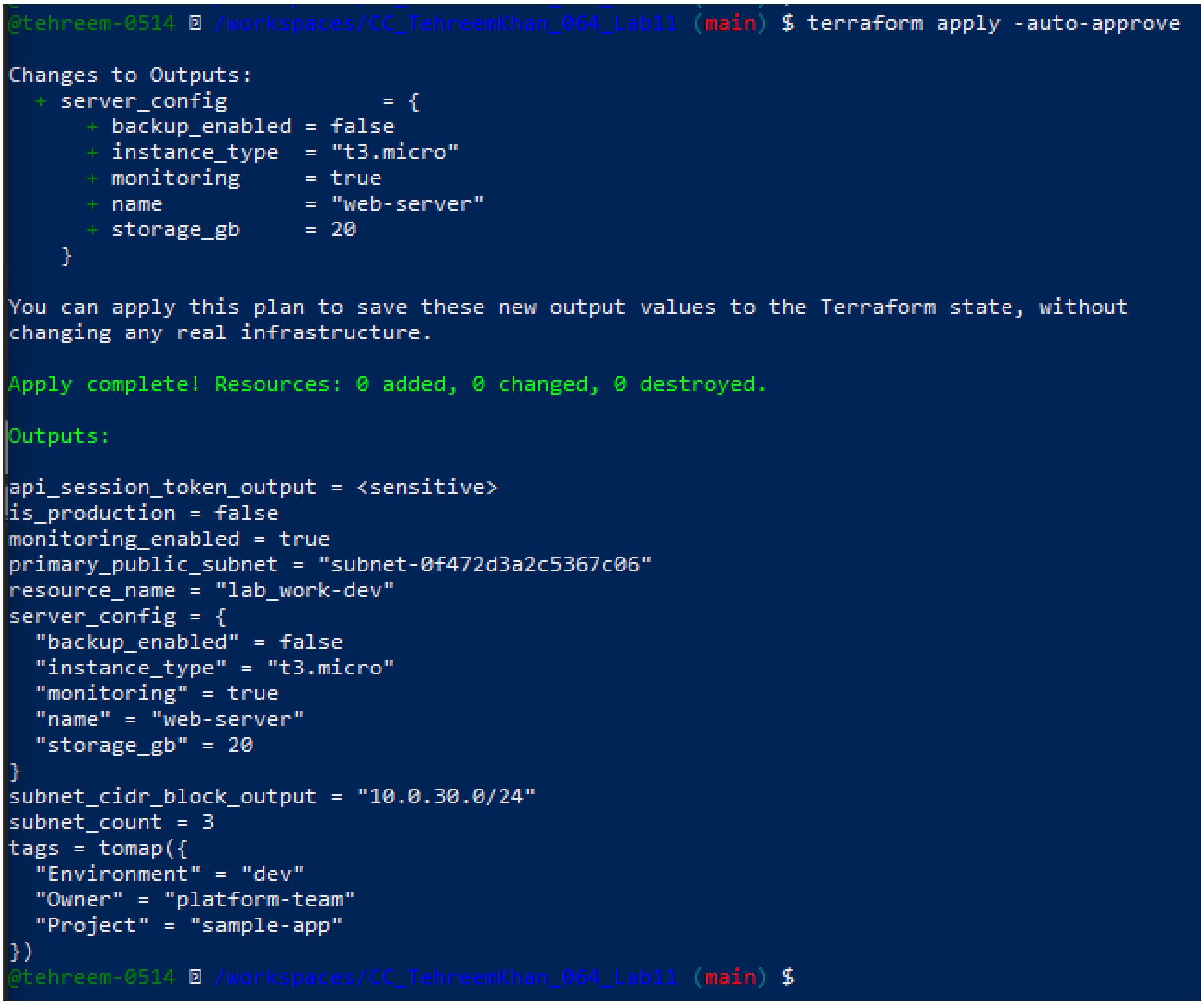
Task 4:



```
variable "tags" {
  type = map(string)
}

output "tags" {
  value = var.tags
}
```

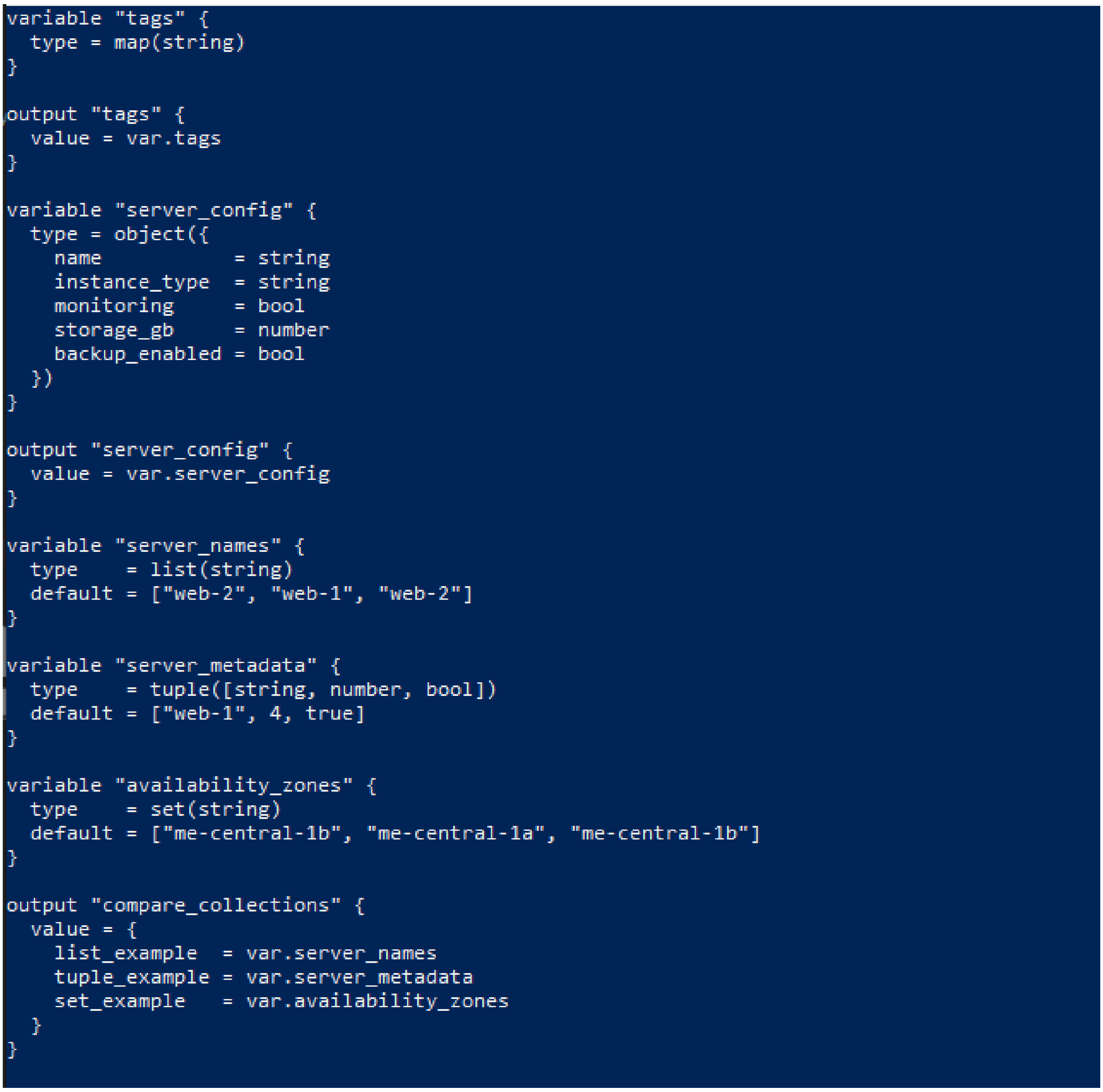


```
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  + tags                        = {
      + Environment = "dev"
      + Owner       = "platform-team"
      + Project     = "sample-app"
    }

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

api_session_token_output = <sensitive>
is_production = false
monitoring_enabled = true
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
```

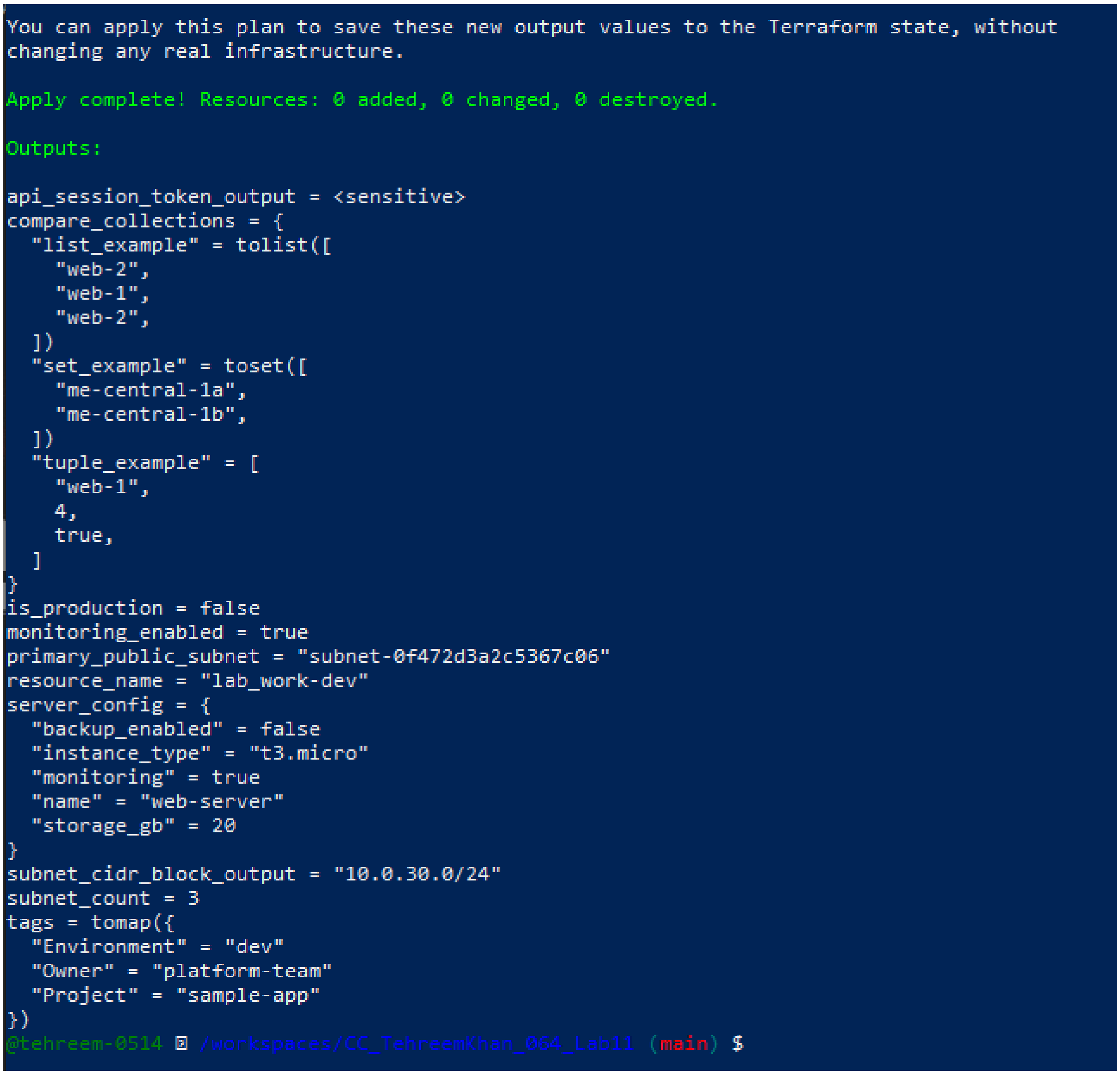


```
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  + server_config              = {
      + backup_enabled = false
      + instance_type  = "t3.micro"
      + monitoring     = true
      + name           = "web-server"
      + storage_gb     = 20
    }

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

api_session_token_output = <sensitive>
is_production = false
monitoring_enabled = true
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

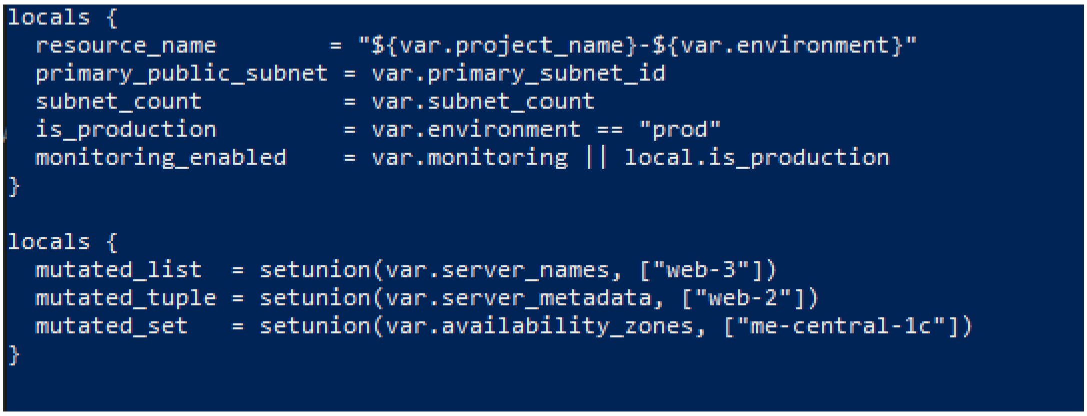
Task 5:



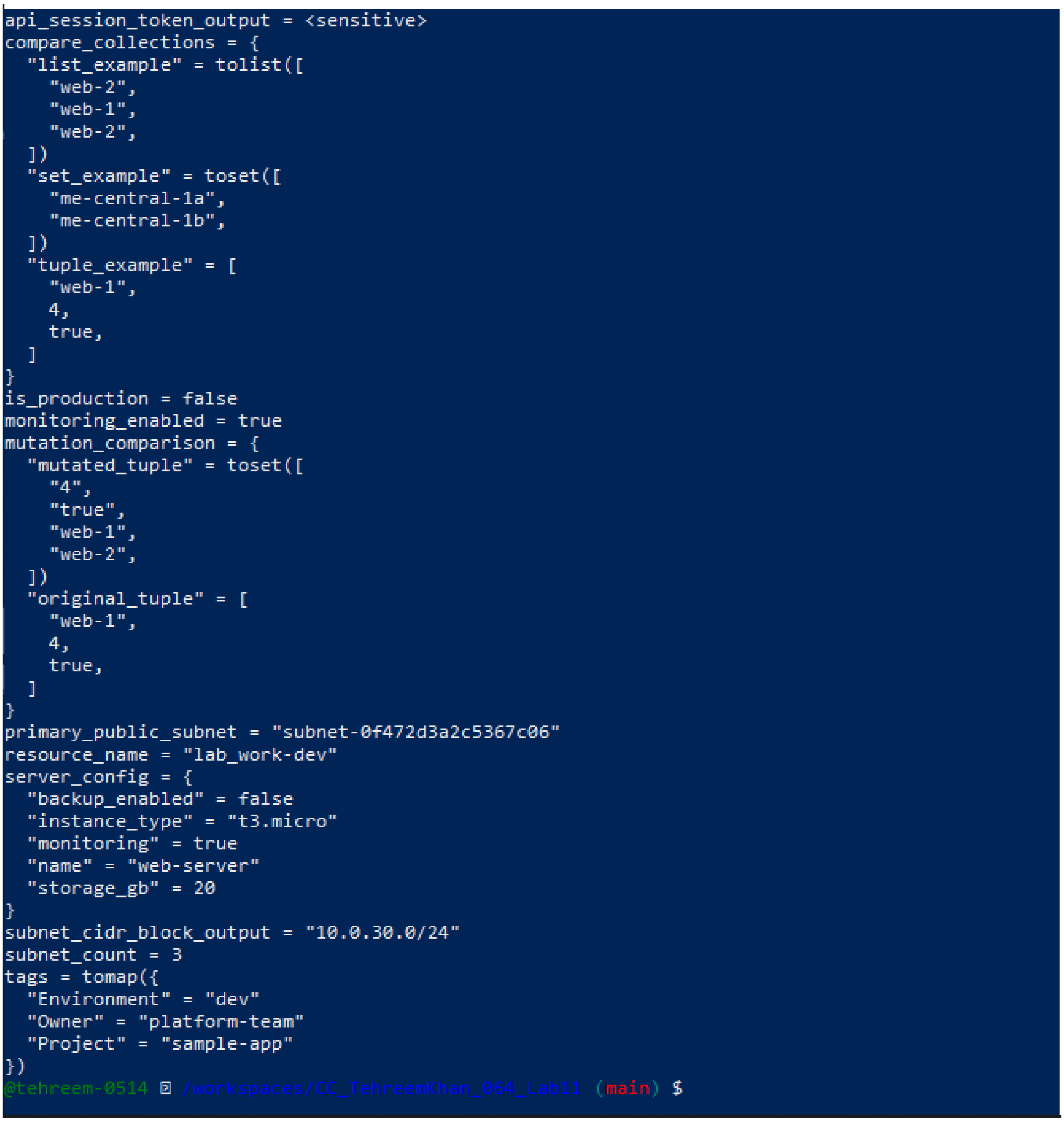
```
variable "tags" {
  type = map(string)
}

output "tags" {
  value = var.tags
}

variable "server_config" {
  type = object({
    name           = string
    instance_type  = string
    monitoring     = bool
    storage_gb     = number
    backup_enabled = bool
  })
}

output "server_config" {
  value = var.server_config
}

variable "server_names" {
  type    = list(string)
  default = ["web-2", "web-1", "web-2"]
}

variable "server_metadata" {
  type    = tuple([string, number, bool])
  default = ["web-1", 4, true]
}

variable "availability_zones" {
  type    = set(string)
  default = ["me-central-1b", "me-central-1a", "me-central-1b"]
}

output "compare_collections" {
  value = {
    list_example  = var.server_names
    tuple_example = var.server_metadata
    set_example   = var.availability_zones
  }
}
```



```
You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

api_session_token_output = <sensitive>
compare_collections = {
  "list_example" = tolist([
    "web-2",
    "web-1",
    "web-2",
  ])
  "set_example" = toset([
    "me-central-1a",
    "me-central-1b",
  ])
  "tuple_example" = [
    "web-1",
    4,
    true,
  ]
}
is_production = false
monitoring_enabled = true
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

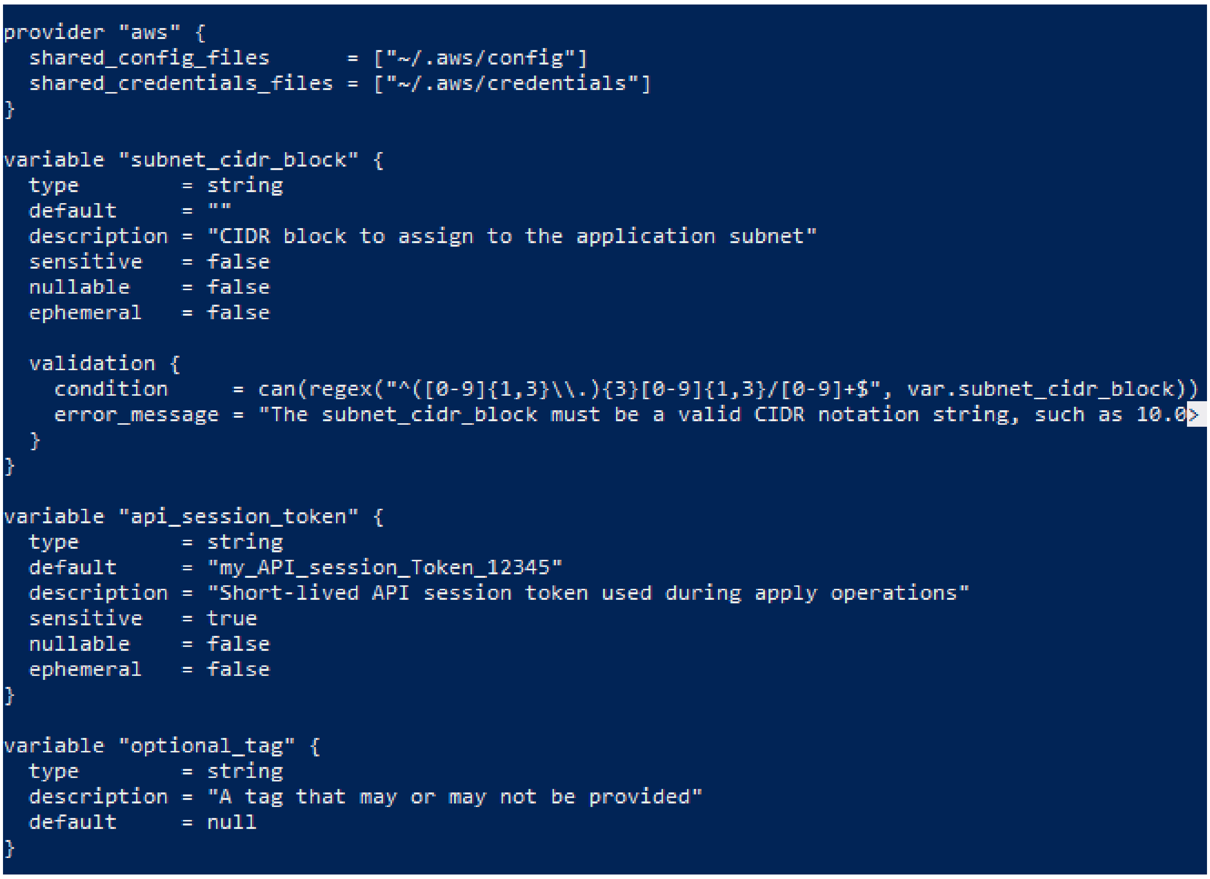


```
locals {
  resource_name         = "${var.project_name}-${var.environment}"
  primary_public_subnet = var.primary_subnet_id
  subnet_count          = var.subnet_count
  is_production         = var.environment == "prod"
  monitoring_enabled    = var.monitoring || local.is_production
}

locals {
  mutated_list  = setunion(var.server_names, ["web-3"])
  mutated_tuple = setunion(var.server_metadata, ["web-2"])
  mutated_set   = setunion(var.availability_zones, ["me-central-1c"])
}
```

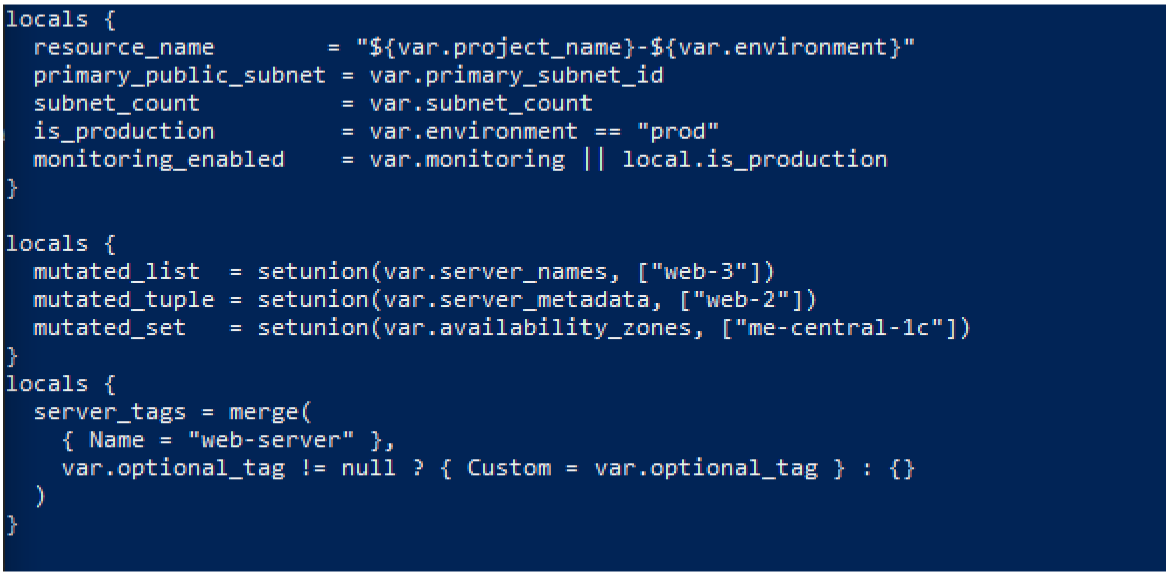


```
api_session_token_output = <sensitive>
compare_collections = {
  "list_example" = tolist([
    "web-2",
    "web-1",
    "web-2",
  ])
  "set_example" = toset([
    "me-central-1a",
    "me-central-1b",
  ])
  "tuple_example" = [
    "web-1",
    4,
    true,
  ]
}
is_production = false
monitoring_enabled = true
mutation_comparison = {
  "mutated_tuple" = toset([
    "4",
    "true",
    "web-1",
    "web-2",
  ])
  "original_tuple" = [
    "web-1",
    4,
    true,
  ]
}
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

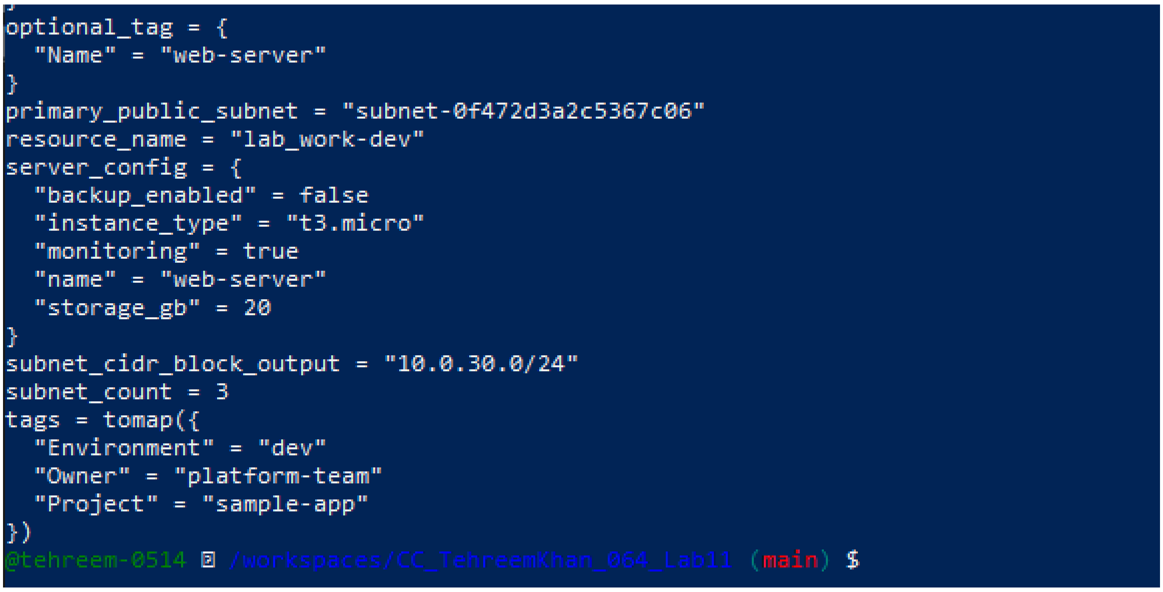
Task 6:



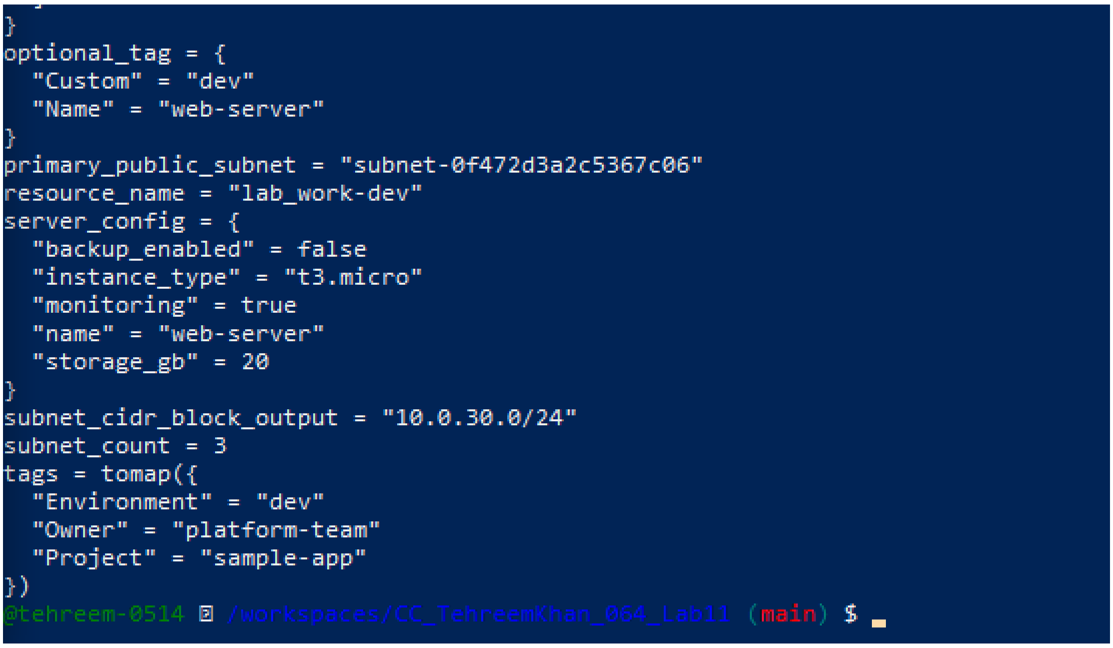
```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

variable "subnet_cidr_block" {
  type        = string
  default     = ""
  description = "CIDR block to assign to the application subnet"
  sensitive   = false
  nullable    = false
  ephemeral   = false

  validation {
    condition     = can(regex("^([0-9]{1,3}\\.){3}[0-9]{1,3}/[0-9]+$", var.subnet_cidr_block))
    error_message = "The subnet_cidr_block must be a valid CIDR notation string, such as 10.0
  }
}

variable "api_session_token" {
  type        = string
  default     = "my_API_session_Token_12345"
  description = "Short-lived API session token used during apply operations"
  sensitive   = true
  nullable    = false
  ephemeral   = false
}

variable "optional_tag" {
  type        = string
  description = "A tag that may or may not be provided"
  default     = null
}
```



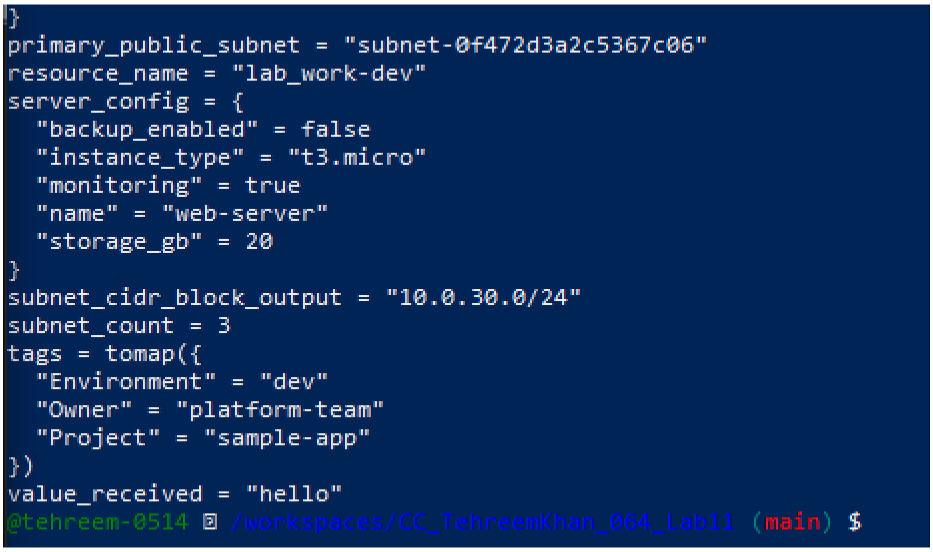
```
locals {
  resource_name         = "${var.project_name}-${var.environment}"
  primary_public_subnet = var.primary_subnet_id
  subnet_count          = var.subnet_count
  is_production         = var.environment == "prod"
  monitoring_enabled    = var.monitoring || local.is_production
}

locals {
  mutated_list  = setunion(var.server_names, ["web-3"])
  mutated_tuple = setunion(var.server_metadata, ["web-2"])
  mutated_set   = setunion(var.availability_zones, ["me-central-1c"])
}
locals {
  server_tags = merge(
    { Name = "web-server" },
    var.optional_tag != null ? { Custom = var.optional_tag } : {}
  )
}
```



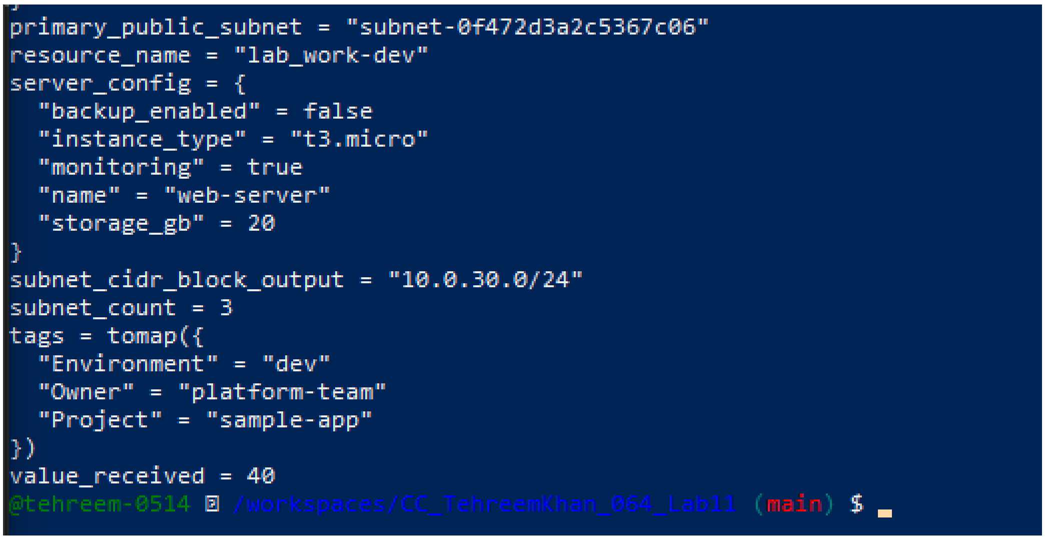
```
optional_tag = {
  "Name" = "web-server"
}
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



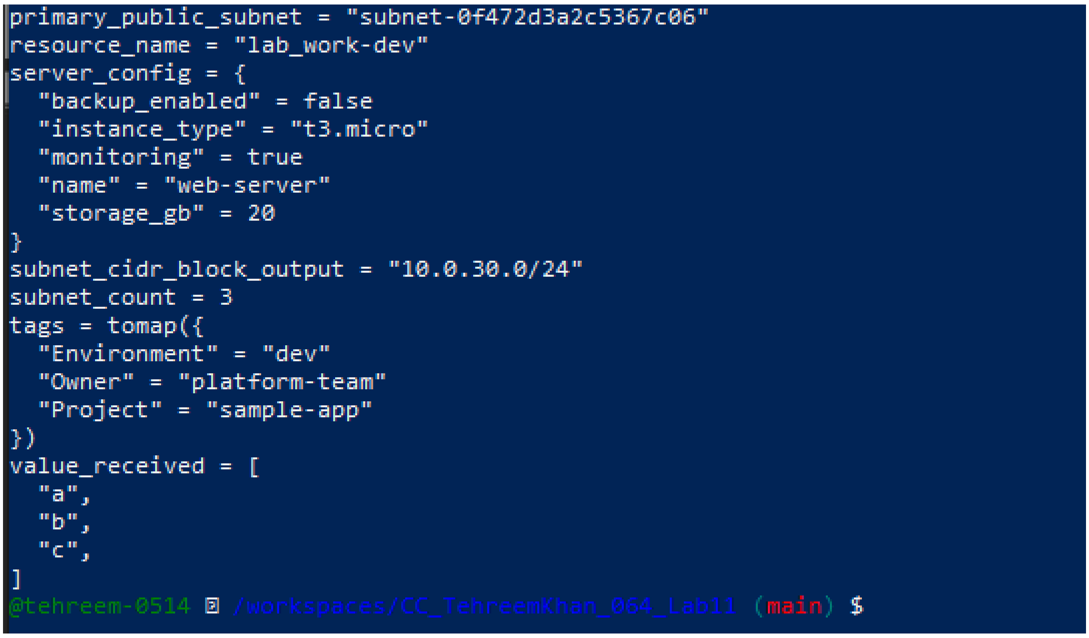
```
}
optional_tag = {
  "Custom" = "dev"
  "Name" = "web-server"
}
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



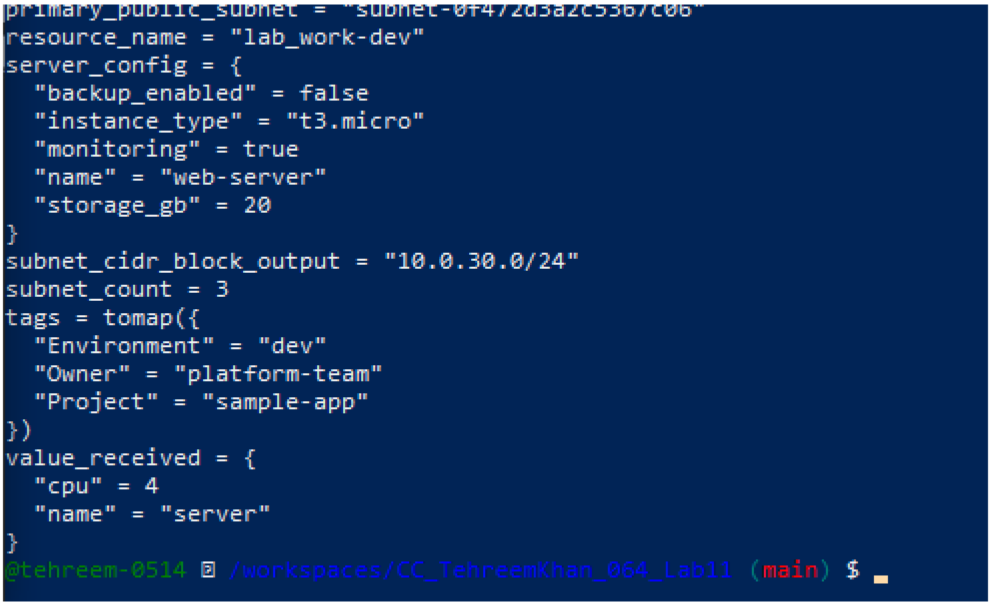
```
}
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
value_received = "hello"
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



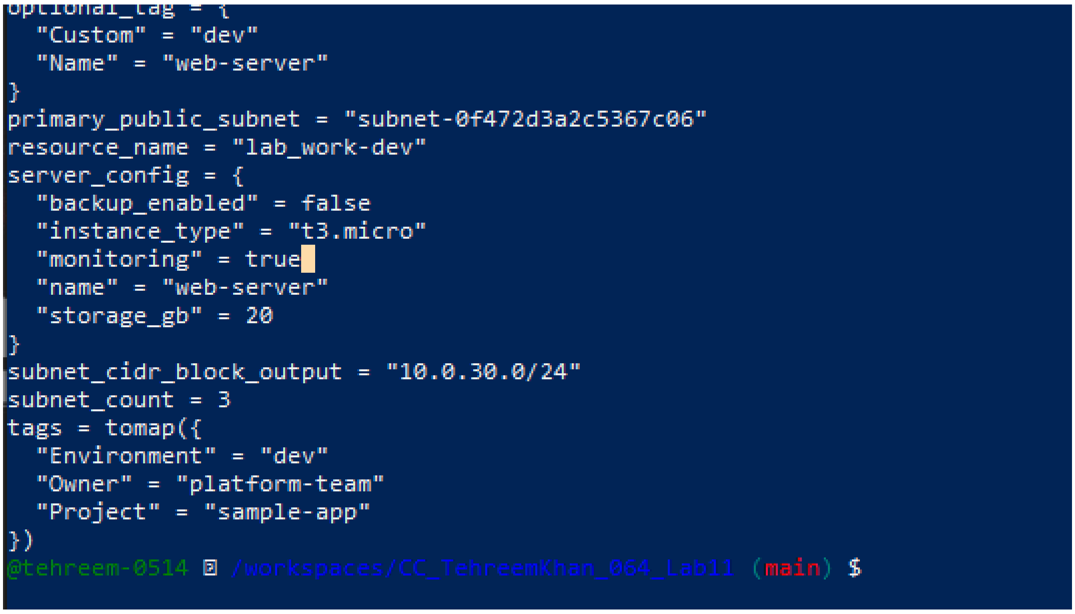
```
}
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
value_received = 40
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ 
```



```
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
value_received = [
  "a",
  "b",
  "c",
]
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

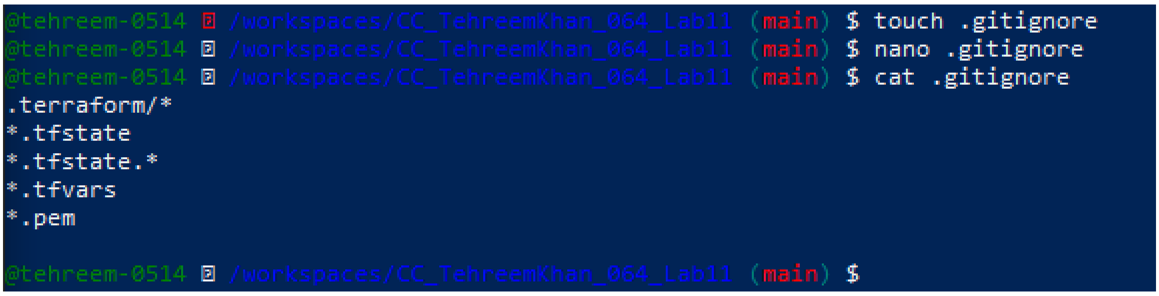


```
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
value_received = {
  "cpu" = 4
  "name" = "server"
}
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ 
```



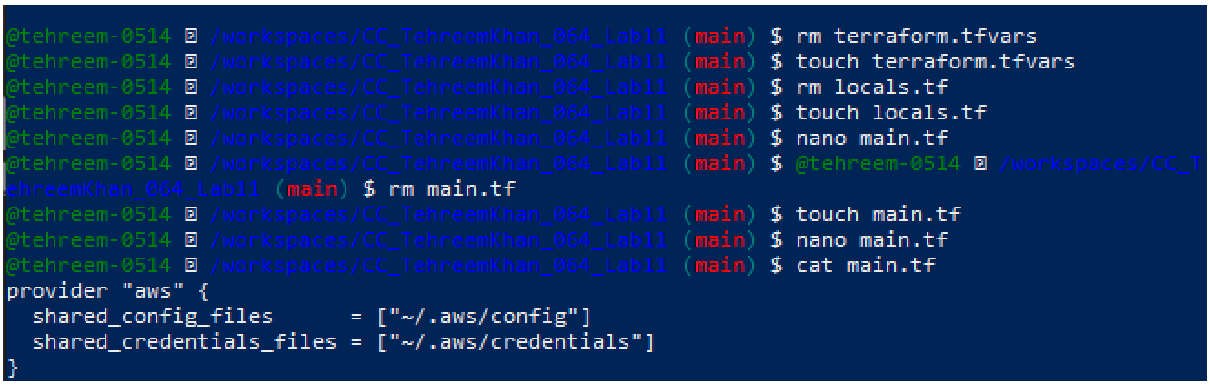
```
optional_tag = {
  "Custom" = "dev"
  "Name" = "web-server"
}
primary_public_subnet = "subnet-0f472d3a2c5367c06"
resource_name = "lab_work-dev"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

Task 7:

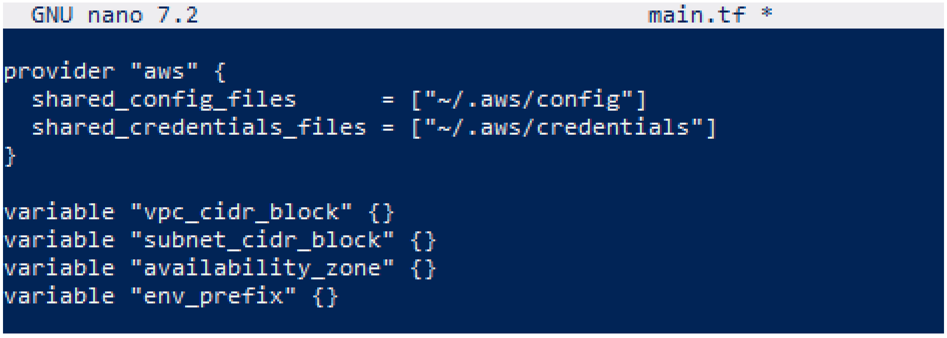


```
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ touch .gitignore
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano .gitignore
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ cat .gitignore
.terraform/*
*.tfstate
*.tfstate.*
*.tfvars
*.pem

@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

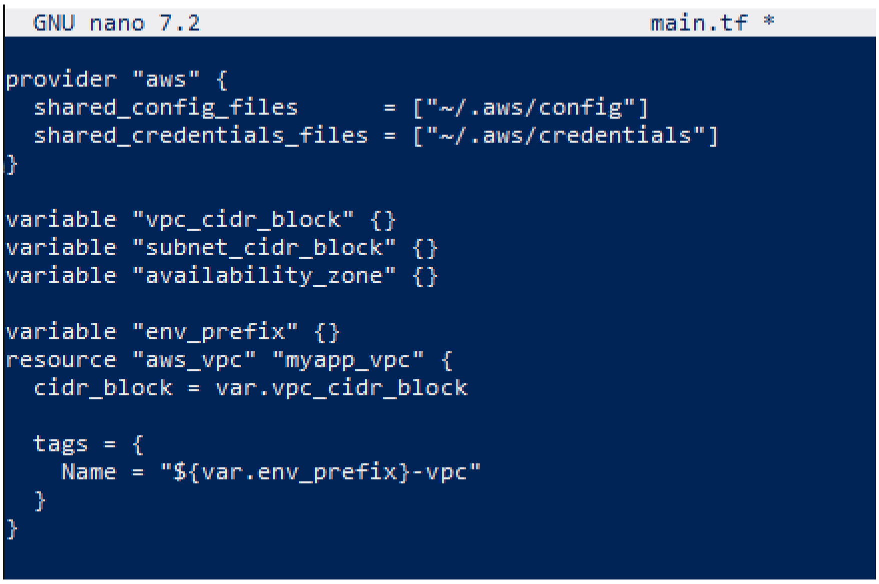
Task 8:



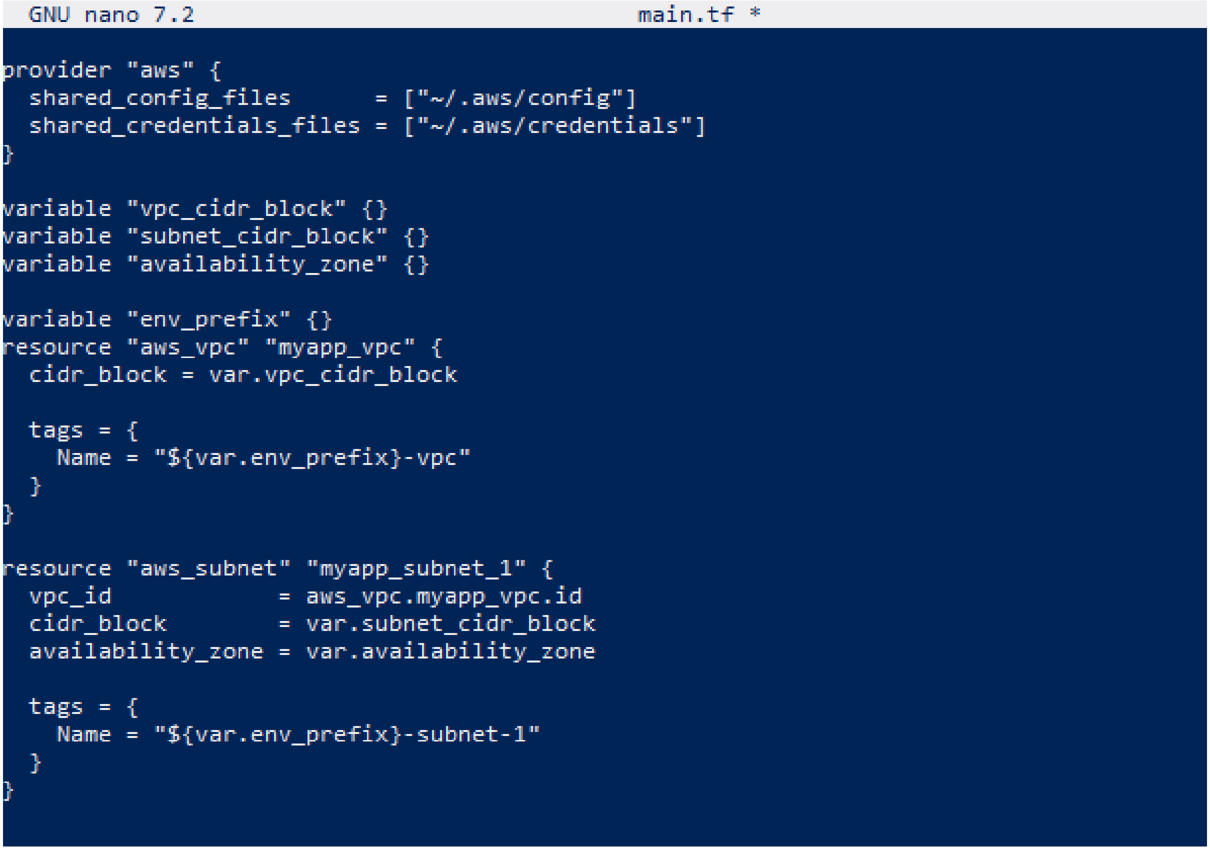
```
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ rm terraform.tfvars
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ touch terraform.tfvars
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ rm locals.tf
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ touch locals.tf
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano main.tf
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ @tehreem-0514 ⊡ /workspaces/CC_T
ehreemKhan_064_Lab11 (main) $ rm main.tf
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ touch main.tf
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano main.tf
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ cat main.tf
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}
```



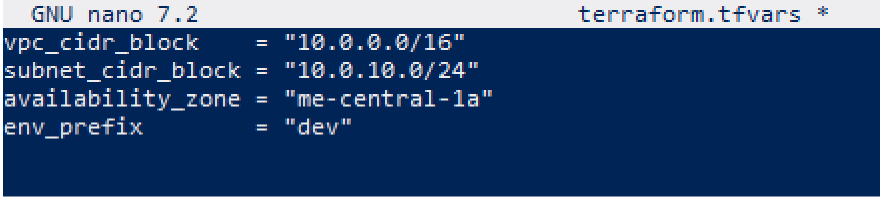
```
  GNU nano 7.2                                   main.tf *

provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

variable "vpc_cidr_block" {}
variable "subnet_cidr_block" {}
variable "availability_zone" {}
variable "env_prefix" {}
```



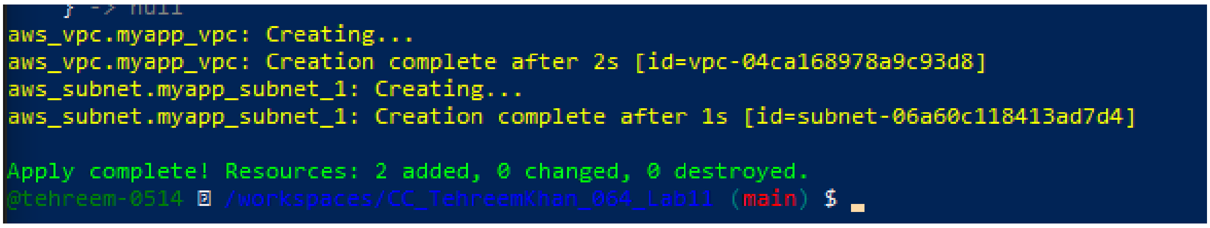
```
  GNU nano 7.2                                   main.tf *

provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

variable "vpc_cidr_block" {}
variable "subnet_cidr_block" {}
variable "availability_zone" {}

variable "env_prefix" {}
resource "aws_vpc" "myapp_vpc" {
  cidr_block = var.vpc_cidr_block

  tags = {
    Name = "${var.env_prefix}-vpc"
  }
}
```



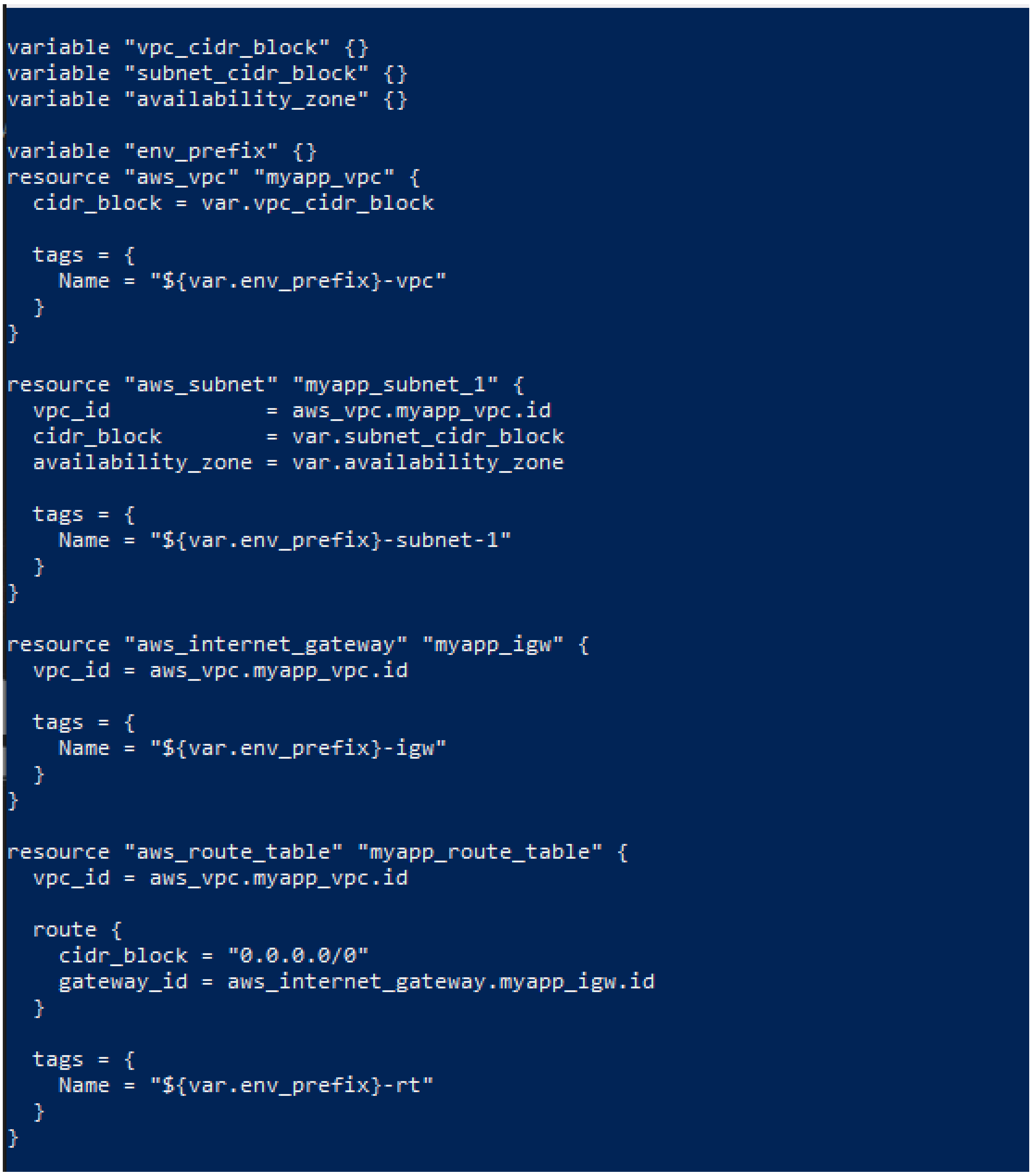
```
  GNU nano 7.2                                        main.tf *

provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

variable "vpc_cidr_block" {}
variable "subnet_cidr_block" {}
variable "availability_zone" {}

variable "env_prefix" {}
resource "aws_vpc" "myapp_vpc" {
  cidr_block = var.vpc_cidr_block

  tags = {
    Name = "${var.env_prefix}-vpc"
  }
}

resource "aws_subnet" "myapp_subnet_1" {
  vpc_id            = aws_vpc.myapp_vpc.id
  cidr_block        = var.subnet_cidr_block
  availability_zone = var.availability_zone

  tags = {
    Name = "${var.env_prefix}-subnet-1"
  }
}
```



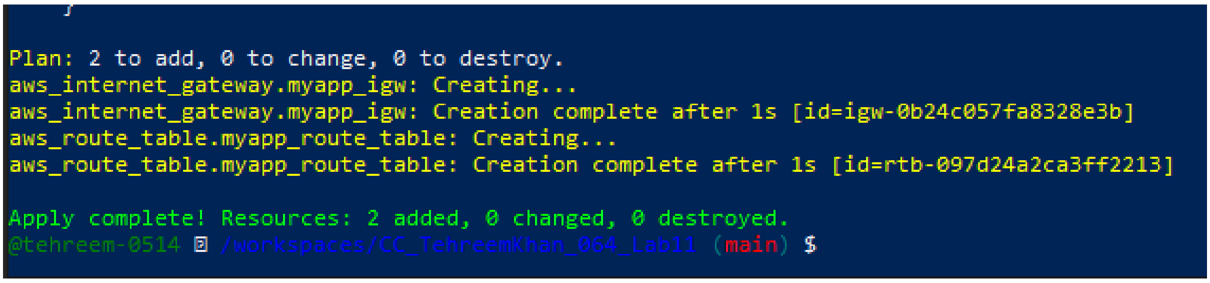
```
  GNU nano 7.2                              terraform.tfvars *
vpc_cidr_block    = "10.0.0.0/16"
subnet_cidr_block = "10.0.10.0/24"
availability_zone = "me-central-1a"
env_prefix        = "dev"
```



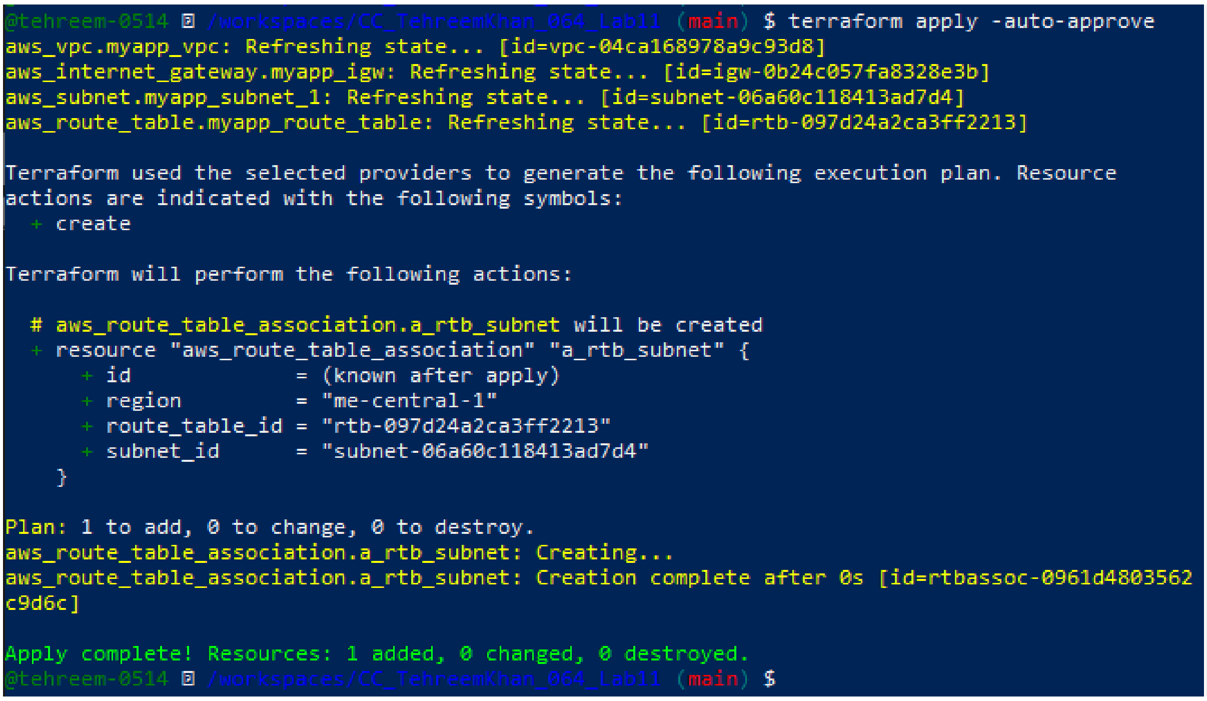
```
    } -> null
aws_vpc.myapp_vpc: Creating...
aws_vpc.myapp_vpc: Creation complete after 2s [id=vpc-04ca168978a9c93d8]
aws_subnet.myapp_subnet_1: Creating...
aws_subnet.myapp_subnet_1: Creation complete after 1s [id=subnet-06a60c118413ad7d4]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



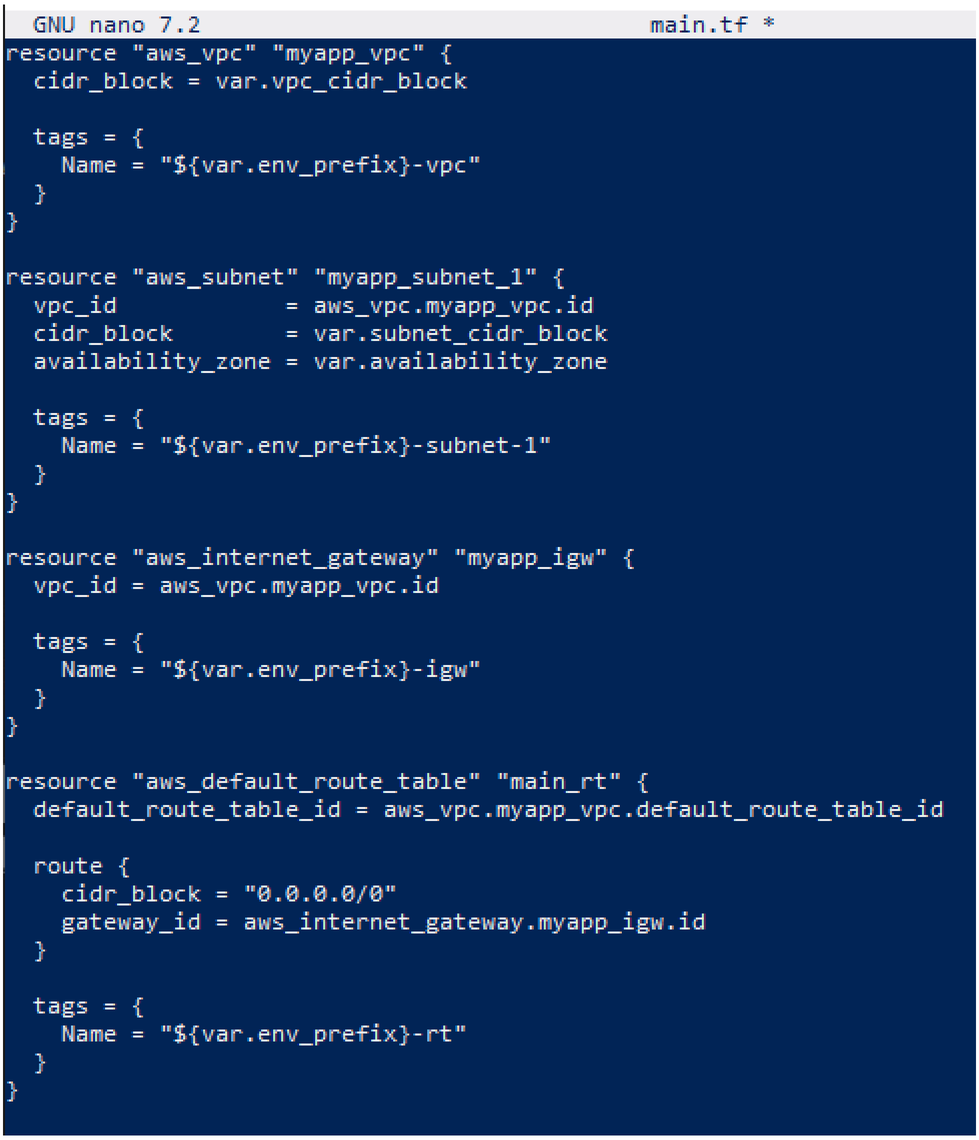
```
variable "vpc_cidr_block" {}
variable "subnet_cidr_block" {}
variable "availability_zone" {}

variable "env_prefix" {}
resource "aws_vpc" "myapp_vpc" {
  cidr_block = var.vpc_cidr_block

  tags = {
    Name = "${var.env_prefix}-vpc"
  }
}

resource "aws_subnet" "myapp_subnet_1" {
  vpc_id            = aws_vpc.myapp_vpc.id
  cidr_block        = var.subnet_cidr_block
  availability_zone = var.availability_zone

  tags = {
    Name = "${var.env_prefix}-subnet-1"
  }
}

resource "aws_internet_gateway" "myapp_igw" {
  vpc_id = aws_vpc.myapp_vpc.id

  tags = {
    Name = "${var.env_prefix}-igw"
  }
}

resource "aws_route_table" "myapp_route_table" {
  vpc_id = aws_vpc.myapp_vpc.id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.myapp_igw.id
  }

  tags = {
    Name = "${var.env_prefix}-rt"
  }
}
```



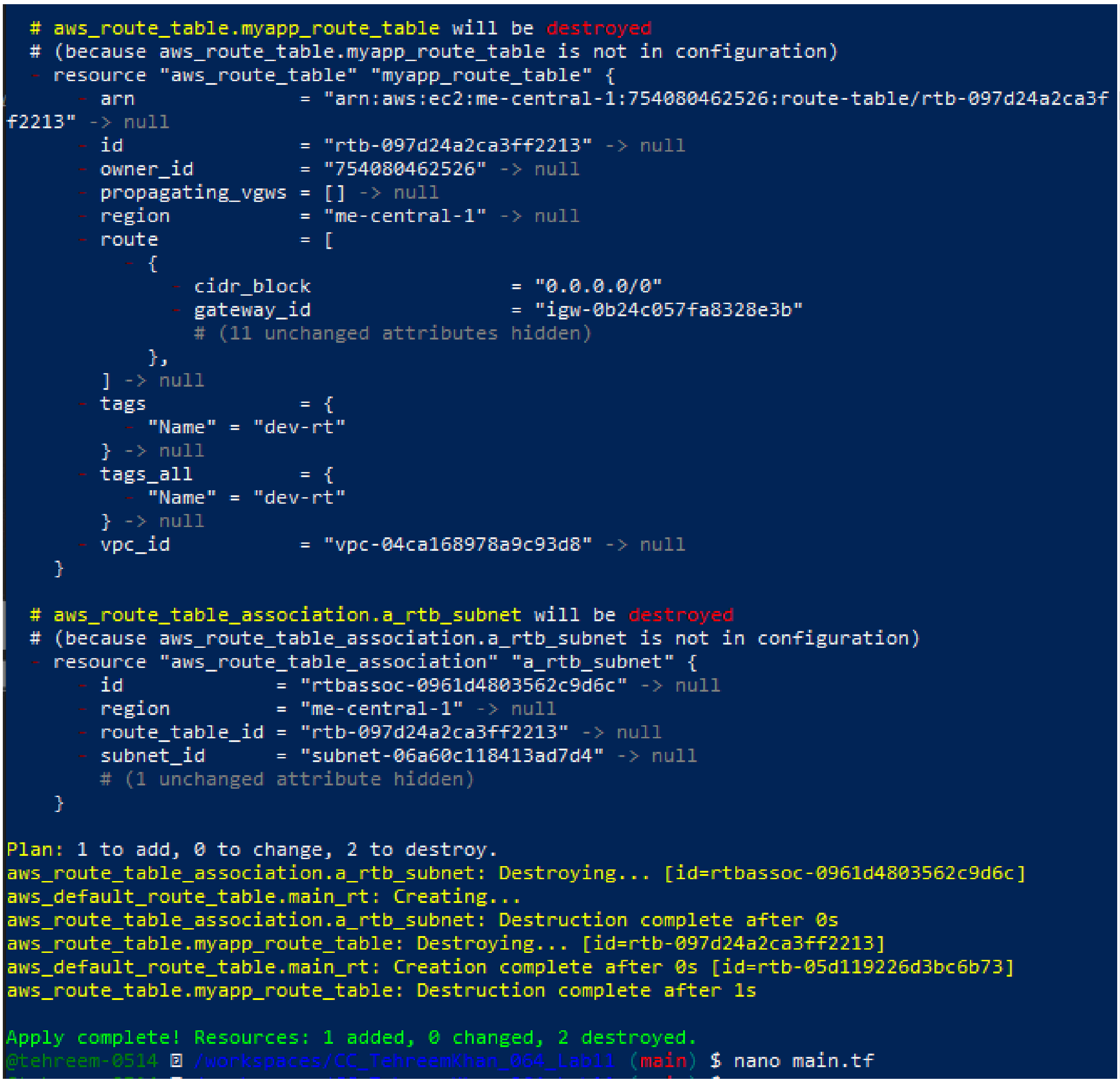
```
    }

Plan: 2 to add, 0 to change, 0 to destroy.
aws_internet_gateway.myapp_igw: Creating...
aws_internet_gateway.myapp_igw: Creation complete after 1s [id=igw-0b24c057fa8328e3b]
aws_route_table.myapp_route_table: Creating...
aws_route_table.myapp_route_table: Creation complete after 1s [id=rtb-097d24a2ca3ff2213]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
@tehreem-0514 ⊠ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



```
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-04ca168978a9c93d8]
aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-0b24c057fa8328e3b]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-06a60c118413ad7d4]
aws_route_table.myapp_route_table: Refreshing state... [id=rtb-097d24a2ca3ff2213]

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_route_table_association.a_rtb_subnet will be created
  + resource "aws_route_table_association" "a_rtb_subnet" {
      + id             = (known after apply)
      + region         = "me-central-1"
      + route_table_id = "rtb-097d24a2ca3ff2213"
      + subnet_id      = "subnet-06a60c118413ad7d4"
    }

Plan: 1 to add, 0 to change, 0 to destroy.
aws_route_table_association.a_rtb_subnet: Creating...
aws_route_table_association.a_rtb_subnet: Creation complete after 0s [id=rtbassoc-0961d4803562
c9d6c]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

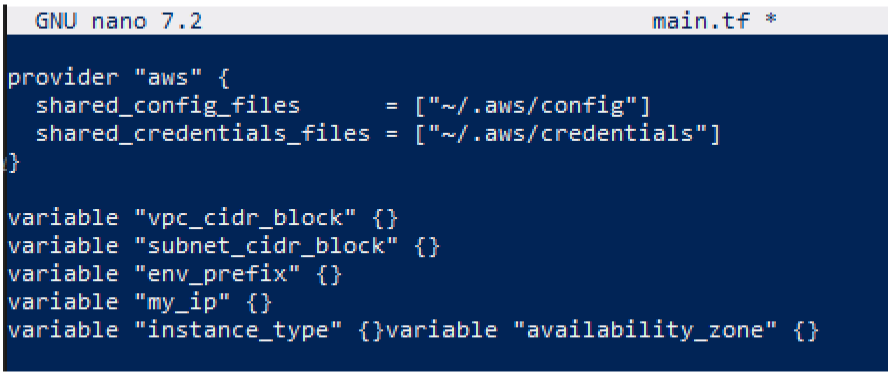


```
  GNU nano 7.2                                          main.tf *
resource "aws_vpc" "myapp_vpc" {
  cidr_block = var.vpc_cidr_block

  tags = {
    Name = "${var.env_prefix}-vpc"
  }
}

resource "aws_subnet" "myapp_subnet_1" {
  vpc_id            = aws_vpc.myapp_vpc.id
  cidr_block        = var.subnet_cidr_block
  availability_zone = var.availability_zone

  tags = {
    Name = "${var.env_prefix}-subnet-1"
  }
}

resource "aws_internet_gateway" "myapp_igw" {
  vpc_id = aws_vpc.myapp_vpc.id

  tags = {
    Name = "${var.env_prefix}-igw"
  }
}

resource "aws_default_route_table" "main_rt" {
  default_route_table_id = aws_vpc.myapp_vpc.default_route_table_id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.myapp_igw.id
  }

  tags = {
    Name = "${var.env_prefix}-rt"
  }
}
```

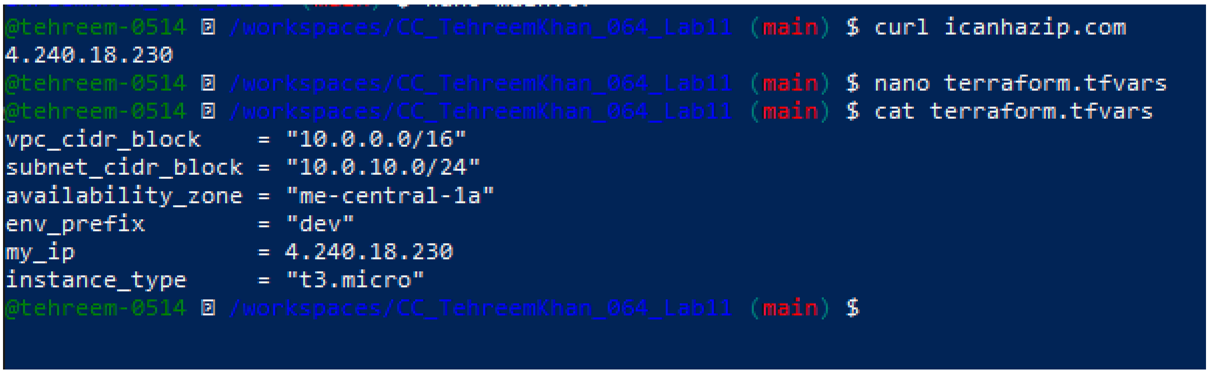


```
  # aws_route_table.myapp_route_table will be destroyed
  # (because aws_route_table.myapp_route_table is not in configuration)
  - resource "aws_route_table" "myapp_route_table" {
      - arn              = "arn:aws:ec2:me-central-1:754080462526:route-table/rtb-097d24a2ca3f
f2213" -> null
      - id               = "rtb-097d24a2ca3ff2213" -> null
      - owner_id         = "754080462526" -> null
      - propagating_vgws = [] -> null
      - region           = "me-central-1" -> null
      - route            = [
          - {
              - cidr_block                 = "0.0.0.0/0"
              - gateway_id                 = "igw-0b24c057fa8328e3b"
                # (11 unchanged attributes hidden)
            },
        ] -> null
      - tags             = {
          - "Name" = "dev-rt"
        } -> null
      - tags_all         = {
          - "Name" = "dev-rt"
        } -> null
      - vpc_id           = "vpc-04ca168978a9c93d8" -> null
    }

  # aws_route_table_association.a_rtb_subnet will be destroyed
  # (because aws_route_table_association.a_rtb_subnet is not in configuration)
  - resource "aws_route_table_association" "a_rtb_subnet" {
      - id             = "rtbassoc-0961d4803562c9d6c" -> null
      - region         = "me-central-1" -> null
      - route_table_id = "rtb-097d24a2ca3ff2213" -> null
      - subnet_id      = "subnet-06a60c118413ad7d4" -> null
        # (1 unchanged attribute hidden)
    }

Plan: 1 to add, 0 to change, 2 to destroy.
aws_route_table_association.a_rtb_subnet: Destroying... [id=rtbassoc-0961d4803562c9d6c]
aws_default_route_table.main_rt: Creating...
aws_route_table_association.a_rtb_subnet: Destruction complete after 0s
aws_route_table.myapp_route_table: Destroying... [id=rtb-097d24a2ca3ff2213]
aws_default_route_table.main_rt: Creation complete after 0s [id=rtb-05d119226d3bc6b73]
aws_route_table.myapp_route_table: Destruction complete after 1s

Apply complete! Resources: 1 added, 0 changed, 2 destroyed.
@tehreem-0514 ⊠ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano main.tf
```

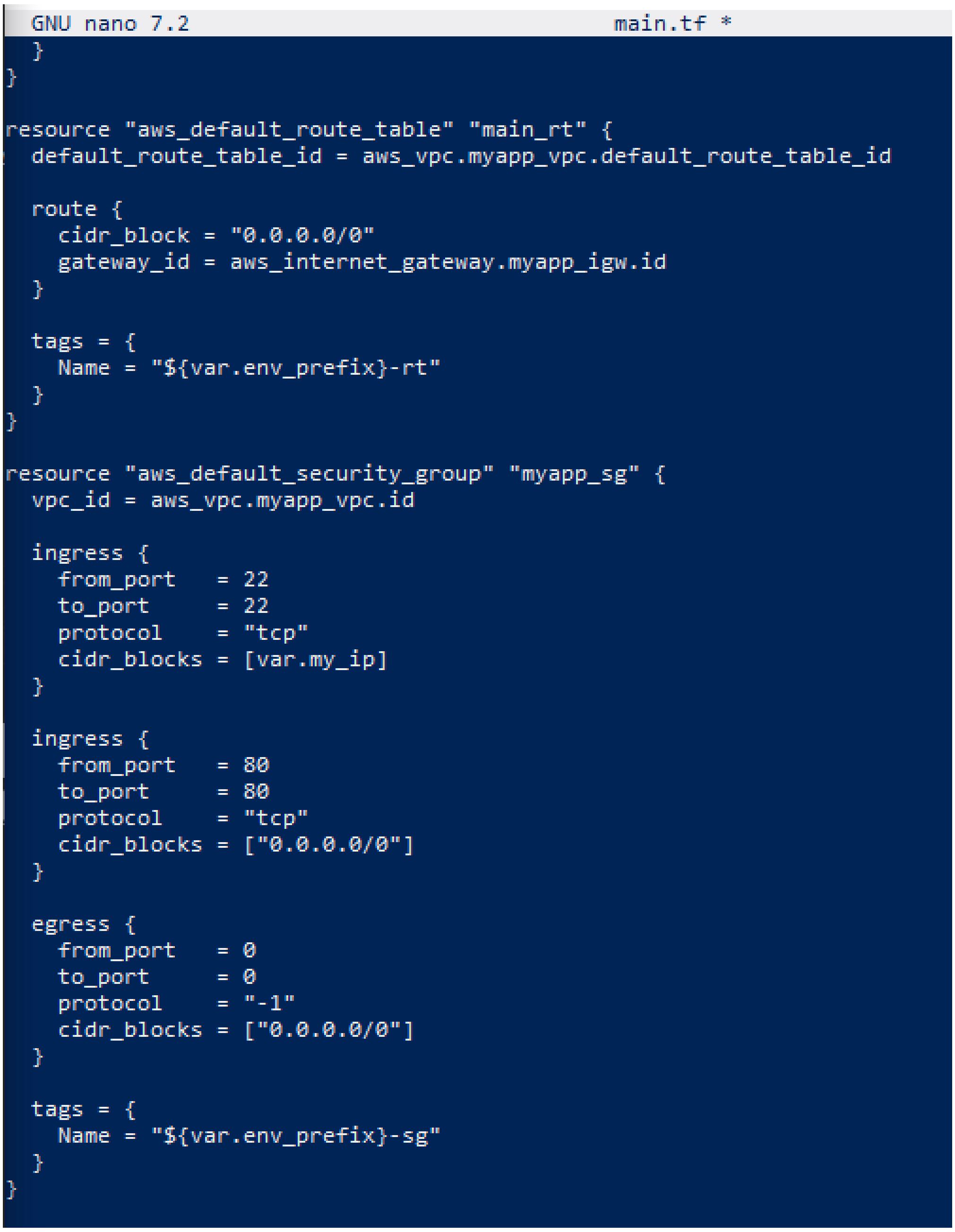
Task 9:



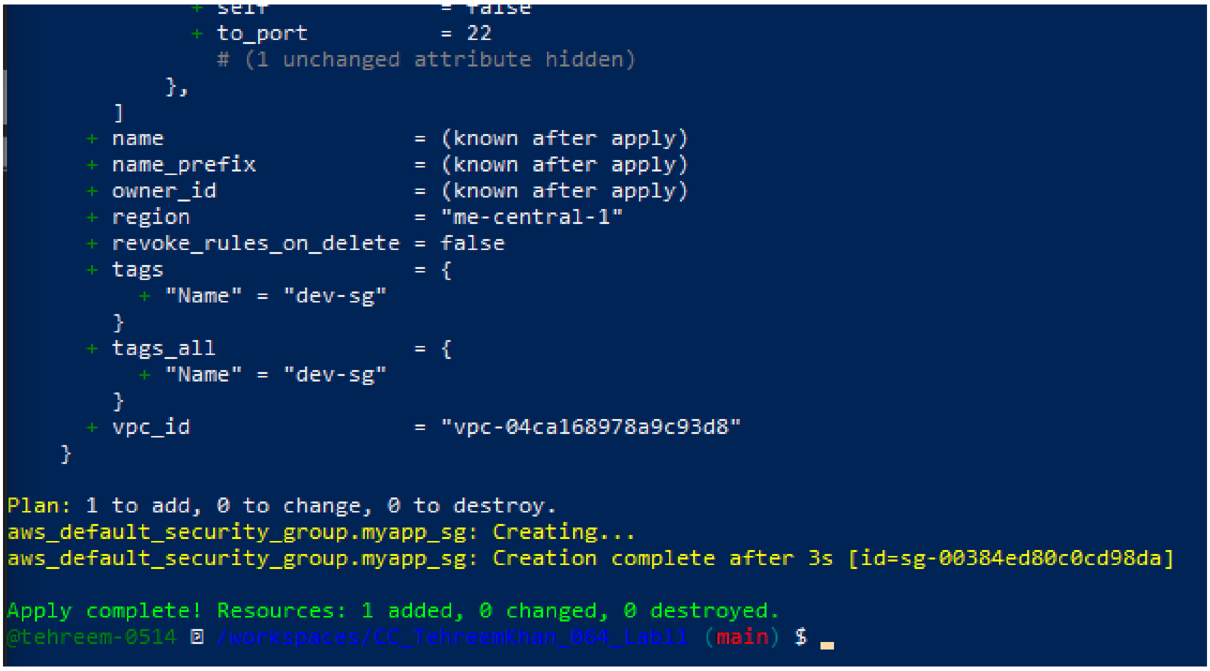
```
  GNU nano 7.2                                    main.tf *

provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

variable "vpc_cidr_block" {}
variable "subnet_cidr_block" {}
variable "env_prefix" {}
variable "my_ip" {}
variable "instance_type" {}variable "availability_zone" {}
```



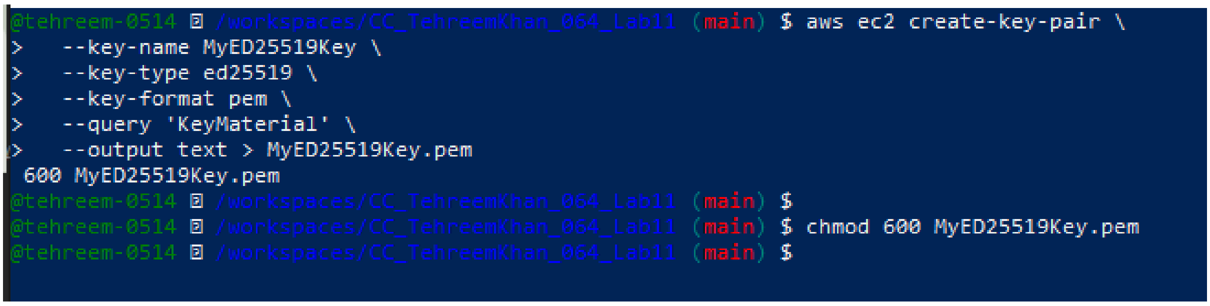
```
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ curl icanhazip.com
4.240.18.230
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano terraform.tfvars
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ cat terraform.tfvars
vpc_cidr_block    = "10.0.0.0/16"
subnet_cidr_block = "10.0.10.0/24"
availability_zone = "me-central-1a"
env_prefix        = "dev"
my_ip             = 4.240.18.230
instance_type     = "t3.micro"
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



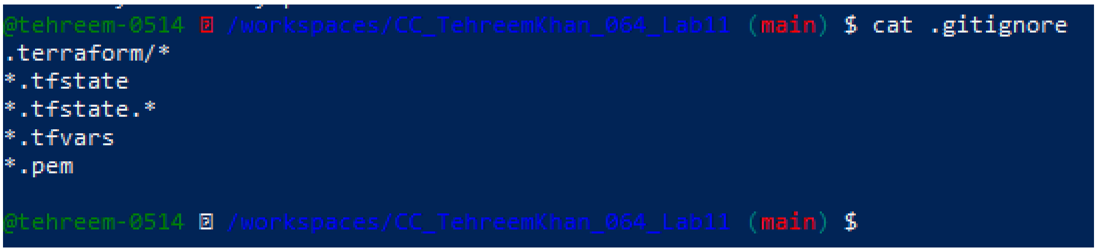
```
  GNU nano 7.2                                          main.tf *
  }
}

resource "aws_default_route_table" "main_rt" {
  default_route_table_id = aws_vpc.myapp_vpc.default_route_table_id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.myapp_igw.id
  }

  tags = {
    Name = "${var.env_prefix}-rt"
  }
}

resource "aws_default_security_group" "myapp_sg" {
  vpc_id = aws_vpc.myapp_vpc.id

  ingress {
    from_port   = 22
    to_port     = 22
    protocol    = "tcp"
    cidr_blocks = [var.my_ip]
  }

  ingress {
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }

  tags = {
    Name = "${var.env_prefix}-sg"
  }
}
```



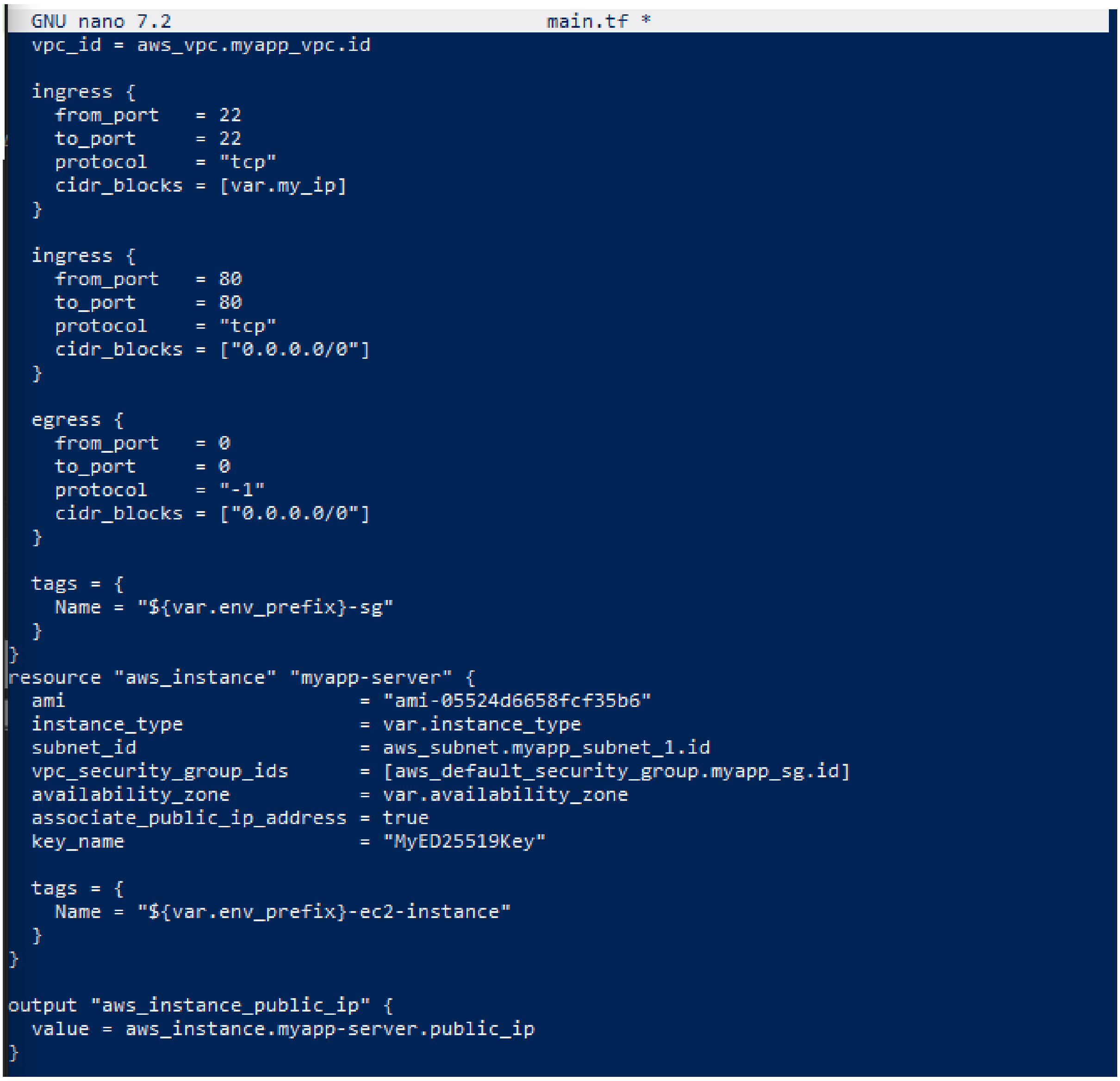
```
            + self              = false
            + to_port           = 22
              # (1 unchanged attribute hidden)
          },
        ]
      + name                   = (known after apply)
      + name_prefix            = (known after apply)
      + owner_id               = (known after apply)
      + region                 = "me-central-1"
      + revoke_rules_on_delete = false
      + tags                   = {
          + "Name" = "dev-sg"
        }
      + tags_all               = {
          + "Name" = "dev-sg"
        }
      + vpc_id                 = "vpc-04ca168978a9c93d8"
    }

Plan: 1 to add, 0 to change, 0 to destroy.
aws_default_security_group.myapp_sg: Creating...
aws_default_security_group.myapp_sg: Creation complete after 3s [id=sg-00384ed80c0cd98da]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ 
```



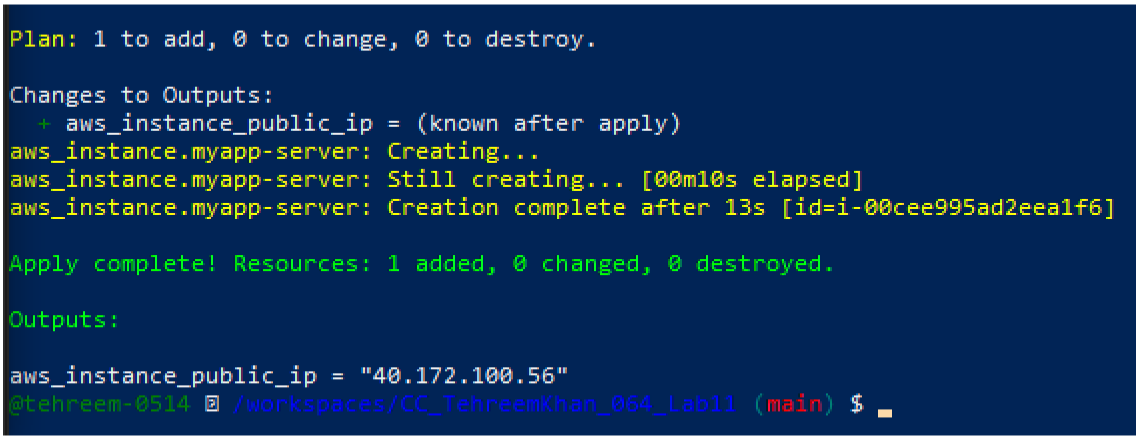
```
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ aws ec2 create-key-pair \
>   --key-name MyED25519Key \
>   --key-type ed25519 \
>   --key-format pem \
>   --query 'KeyMaterial' \
>   --output text > MyED25519Key.pem
 600 MyED25519Key.pem
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ chmod 600 MyED25519Key.pem
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



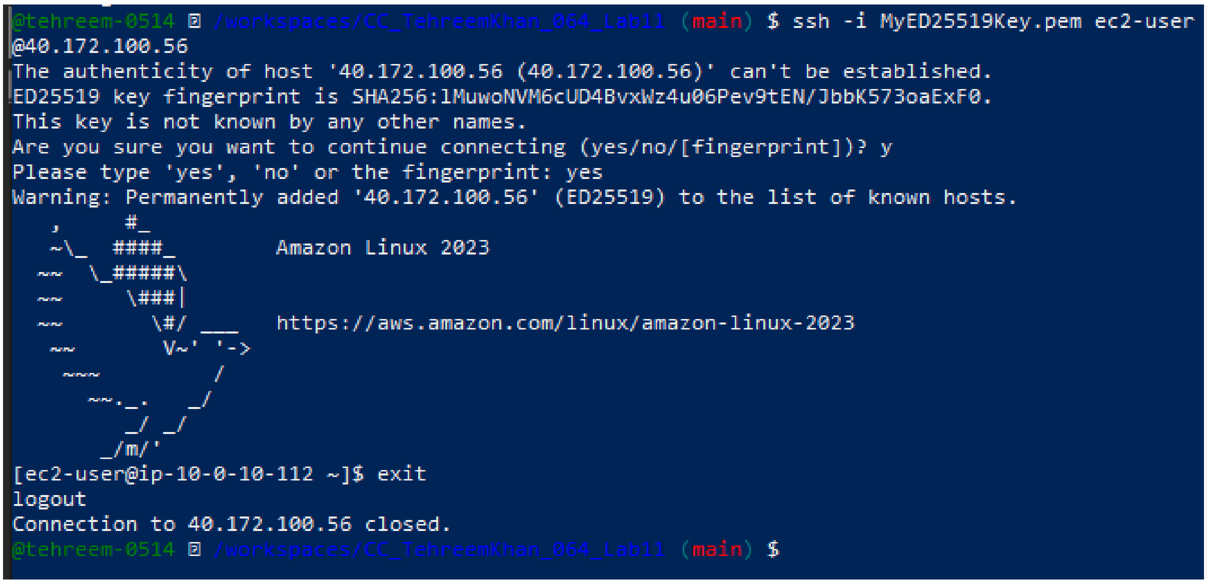
```
@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ cat .gitignore
.terraform/*
*.tfstate
*.tfstate.*
*.tfvars
*.pem

@tehreem-0514 ⊡ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



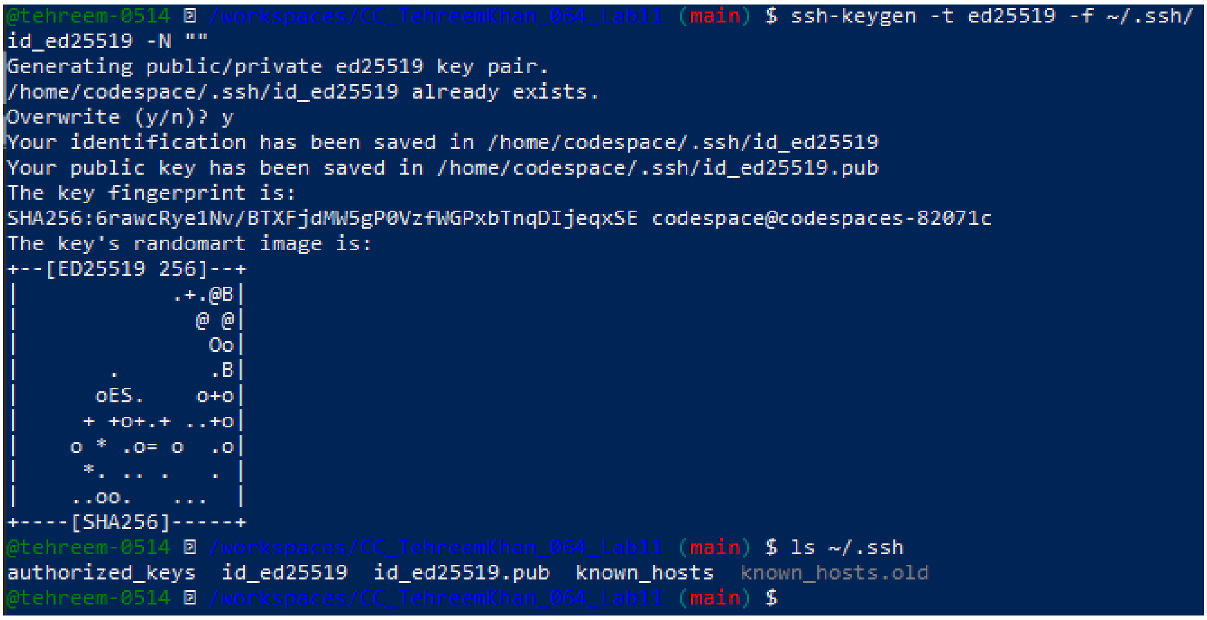
```
  GNU nano 7.2                              main.tf *
  vpc_id = aws_vpc.myapp_vpc.id

  ingress {
    from_port   = 22
    to_port     = 22
    protocol    = "tcp"
    cidr_blocks = [var.my_ip]
  }

  ingress {
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }

  tags = {
    Name = "${var.env_prefix}-sg"
  }
}
resource "aws_instance" "myapp-server" {
  ami                         = "ami-05524d6658fcf35b6"
  instance_type               = var.instance_type
  subnet_id                   = aws_subnet.myapp_subnet_1.id
  vpc_security_group_ids      = [aws_default_security_group.myapp_sg.id]
  availability_zone           = var.availability_zone
  associate_public_ip_address = true
  key_name                    = "MyED25519Key"

  tags = {
    Name = "${var.env_prefix}-ec2-instance"
  }
}

output "aws_instance_public_ip" {
  value = aws_instance.myapp-server.public_ip
}
```



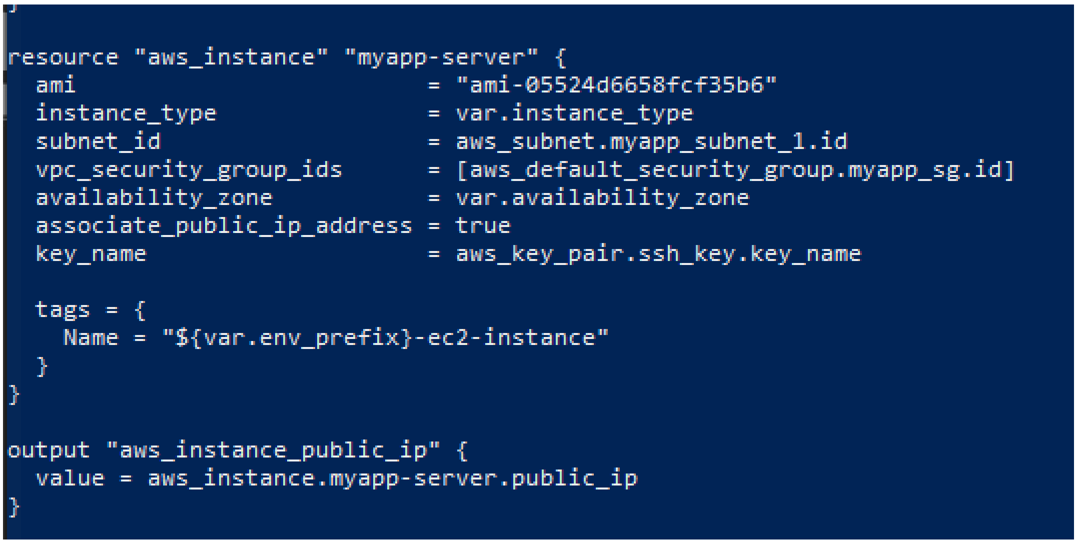
```
Plan: 1 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + aws_instance_public_ip = (known after apply)
aws_instance.myapp-server: Creating...
aws_instance.myapp-server: Still creating... [00m10s elapsed]
aws_instance.myapp-server: Creation complete after 13s [id=i-00cee995ad2eea1f6]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:

aws_instance_public_ip = "40.172.100.56"
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ 
```



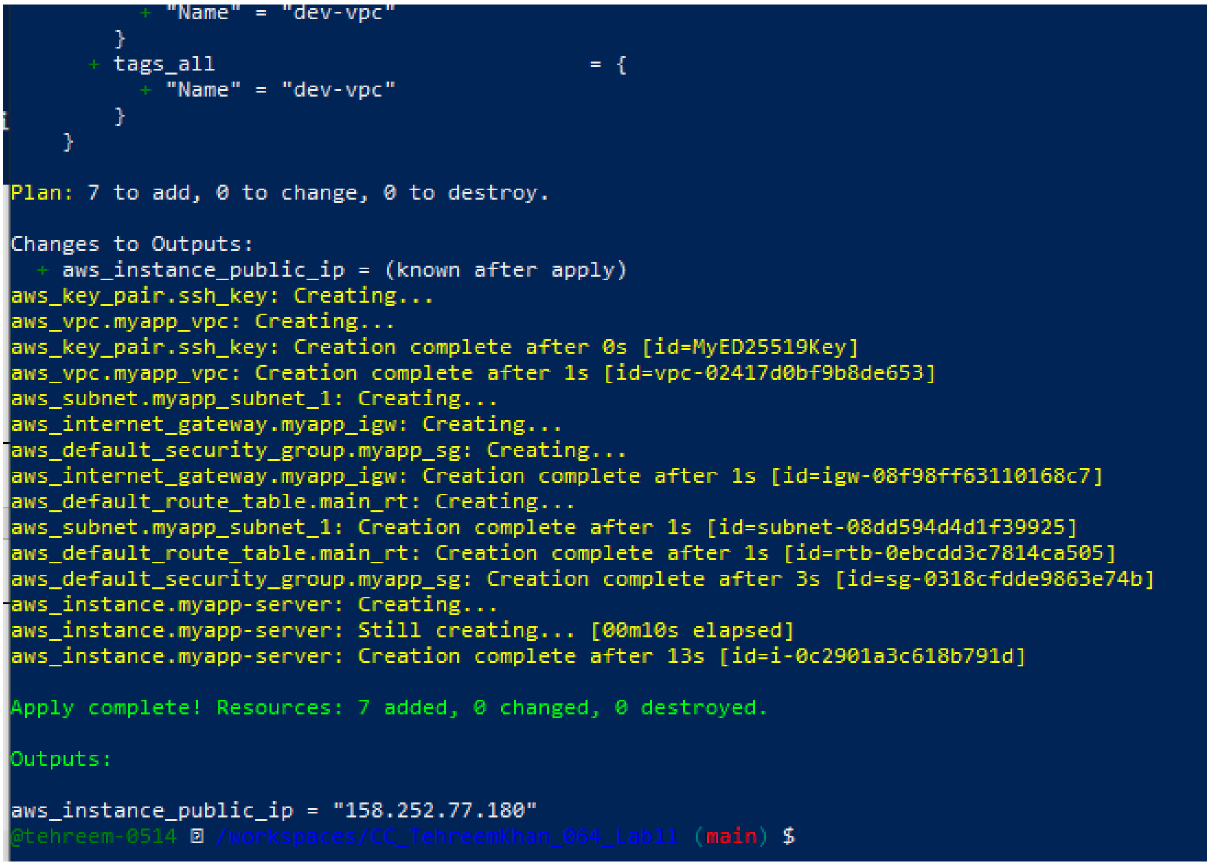
```
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ ssh -i MyED25519Key.pem ec2-user
@40.172.100.56
The authenticity of host '40.172.100.56 (40.172.100.56)' can't be established.
ED25519 key fingerprint is SHA256:1MuwoNVM6cUD4BvxWz4u06Pev9tEN/JbbK573oaExF0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '40.172.100.56' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-10-112 ~]$ exit
logout
Connection to 40.172.100.56 closed.
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



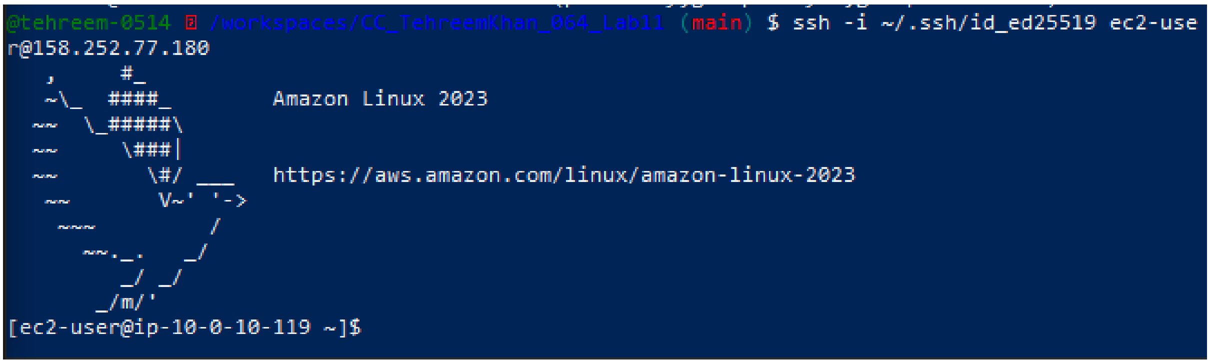
```
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ ssh-keygen -t ed25519 -f ~/.ssh/
id_ed25519 -N ""
Generating public/private ed25519 key pair.
/home/codespace/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Your identification has been saved in /home/codespace/.ssh/id_ed25519
Your public key has been saved in /home/codespace/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:6rawcRye1Nv/BTXFjdMW5gP0VzfWGPxbTnqDIjeqxSE codespace@codespaces-82071c
The key's randomart image is:
+--[ED25519 256]--+
|            .+.@B|
|              @ @|
|               Oo|
|       .       .B|
|      oES.    o+o|
|     + +o+.+ ..+o|
|    o * .o= o  .o|
|     *. .. .   . |
|    ..oo.   ...  |
+----[SHA256]-----+
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ ls ~/.ssh
authorized_keys  id_ed25519  id_ed25519.pub  known_hosts  known_hosts.old
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



```
}

resource "aws_instance" "myapp-server" {
  ami                         = "ami-05524d6658fcf35b6"
  instance_type               = var.instance_type
  subnet_id                   = aws_subnet.myapp_subnet_1.id
  vpc_security_group_ids      = [aws_default_security_group.myapp_sg.id]
  availability_zone           = var.availability_zone
  associate_public_ip_address = true
  key_name                    = aws_key_pair.ssh_key.key_name

  tags = {
    Name = "${var.env_prefix}-ec2-instance"
  }
}

output "aws_instance_public_ip" {
  value = aws_instance.myapp-server.public_ip
}
```

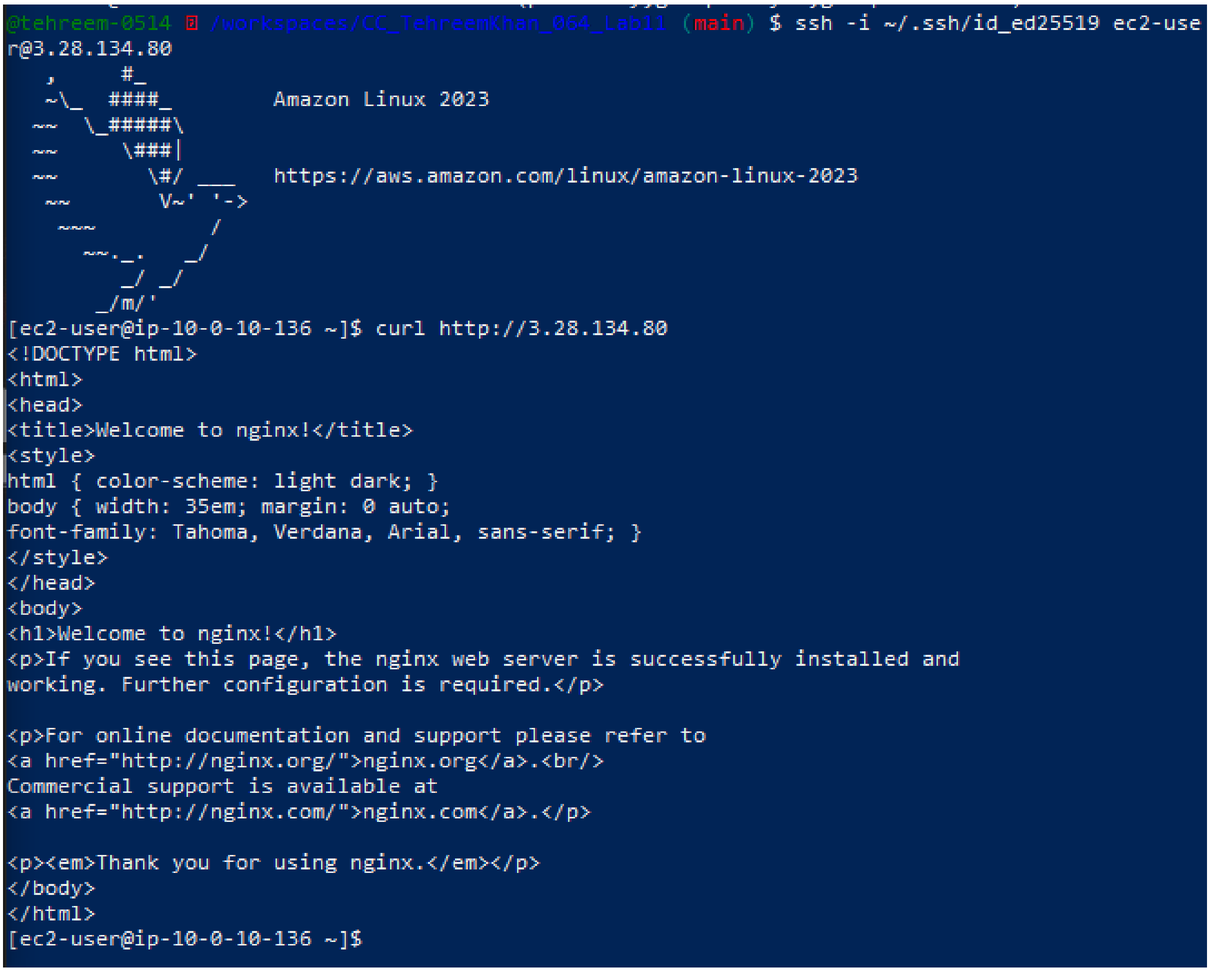


```
          + "Name" = "dev-vpc"
        }
      + tags_all                             = {
          + "Name" = "dev-vpc"
        }
    }

Plan: 7 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + aws_instance_public_ip = (known after apply)
aws_key_pair.ssh_key: Creating...
aws_vpc.myapp_vpc: Creating...
aws_key_pair.ssh_key: Creation complete after 0s [id=MyED25519Key]
aws_vpc.myapp_vpc: Creation complete after 1s [id=vpc-02417d0bf9b8de653]
aws_subnet.myapp_subnet_1: Creating...
aws_internet_gateway.myapp_igw: Creating...
aws_default_security_group.myapp_sg: Creating...
aws_internet_gateway.myapp_igw: Creation complete after 1s [id=igw-08f98ff63110168c7]
aws_default_route_table.main_rt: Creating...
aws_subnet.myapp_subnet_1: Creation complete after 1s [id=subnet-08dd594d4d1f39925]
aws_default_route_table.main_rt: Creation complete after 1s [id=rtb-0ebcdd3c7814ca505]
aws_default_security_group.myapp_sg: Creation complete after 3s [id=sg-0318cfdde9863e74b]
aws_instance.myapp-server: Creating...
aws_instance.myapp-server: Still creating... [00m10s elapsed]
aws_instance.myapp-server: Creation complete after 13s [id=i-0c2901a3c618b791d]

Apply complete! Resources: 7 added, 0 changed, 0 destroyed.

Outputs:

aws_instance_public_ip = "158.252.77.180"
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

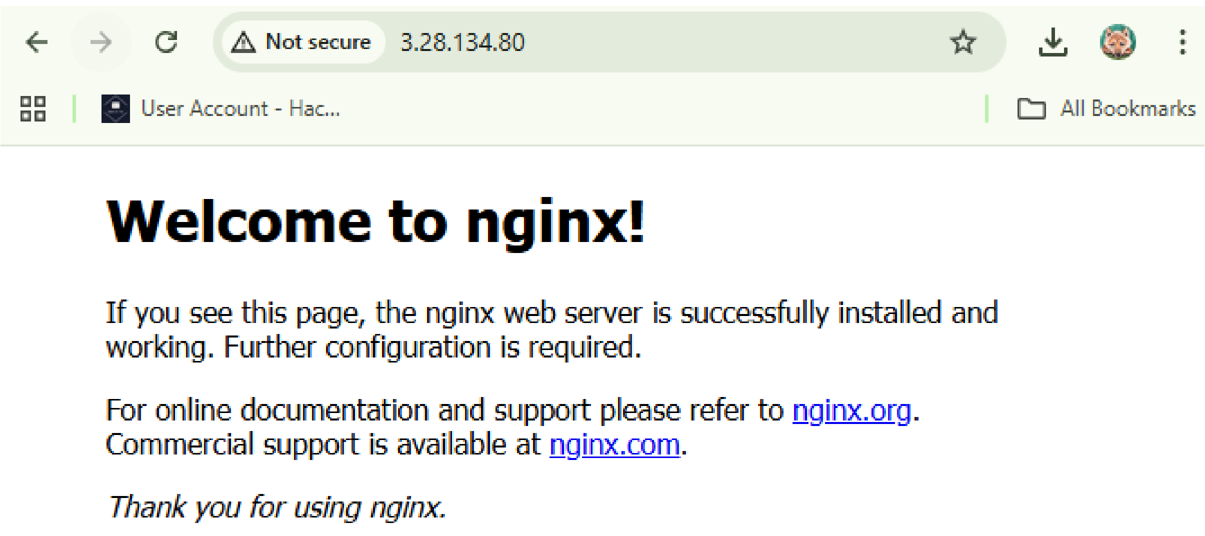


```
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ ssh -i ~/.ssh/id_ed25519 ec2-use
r@158.252.77.180
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-10-119 ~]$
```

I later generated new IP bcz of errors.



```
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ ssh -i ~/.ssh/id_ed25519 ec2-use
r@3.28.134.80
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-10-136 ~]$ curl http://3.28.134.80
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
[ec2-user@ip-10-0-10-136 ~]$
```

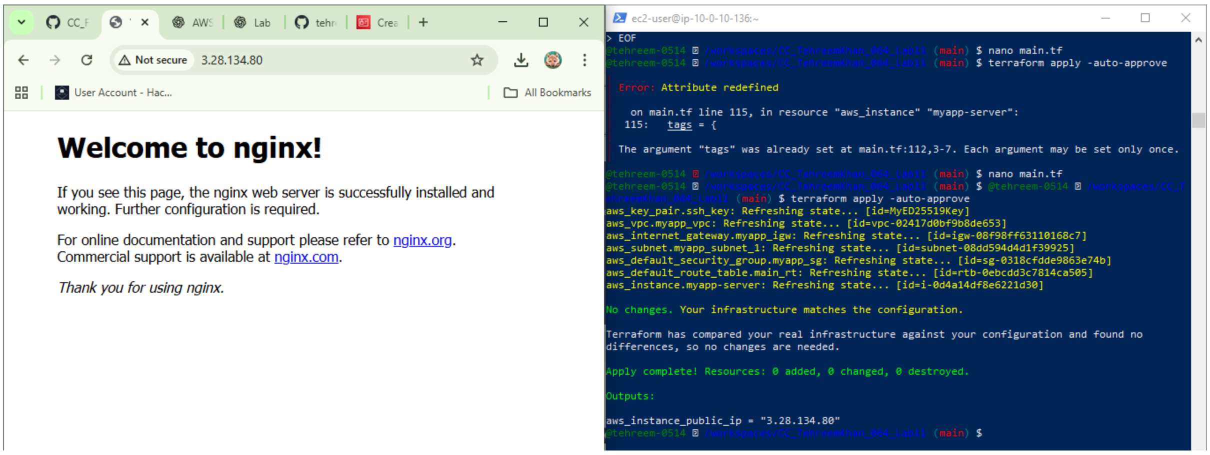


Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org. Commercial support is available at nginx.com.

Thank you for using nginx.



Welcome to nginx!

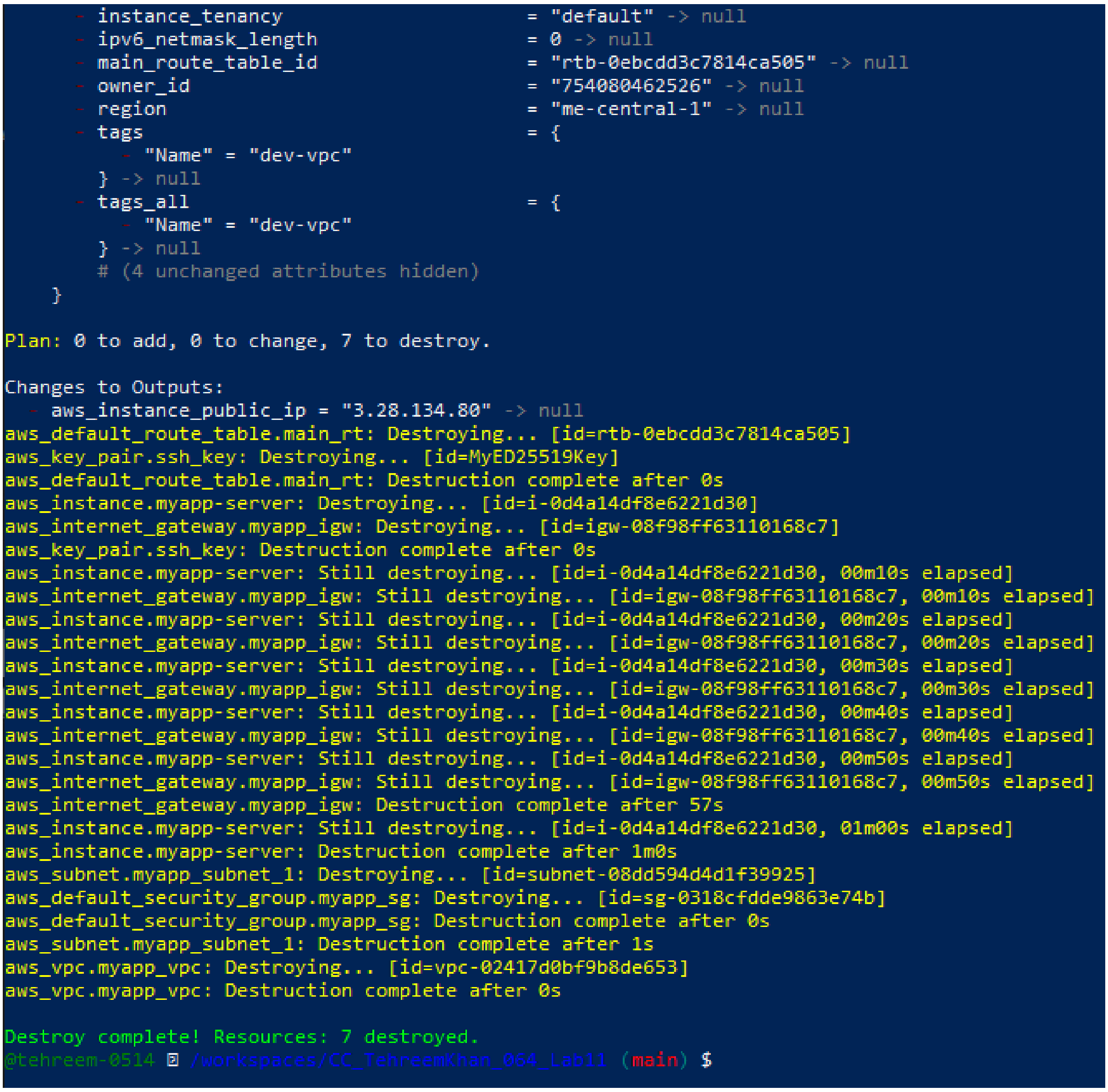
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

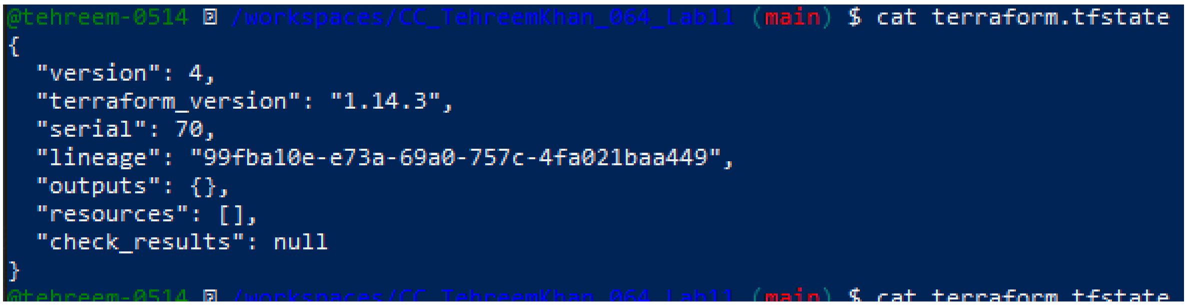
Thank you for using nginx.

```
> EOF
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano main.tf
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve

 Error: Attribute redefined

   on main.tf line 115, in resource "aws_instance" "myapp-server":
  115:   tags = {

 The argument "tags" was already set at main.tf:112,3-7. Each argument may be set only once.

@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ nano main.tf
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ @tehreem-0514 □ /workspaces/CC_T
ehreemKhan_064_Lab11 (main) $ terraform apply -auto-approve
aws_key_pair.ssh_key: Refreshing state... [id=MyED25519Key]
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-02417d0bf9b8de653]
aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-08f98ff63110168c7]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-08dd594d4d1f39925]
aws_default_security_group.myapp_sg: Refreshing state... [id=sg-0318cfdde9863e74b]
aws_default_route_table.main_rt: Refreshing state... [id=rtb-0ebcdd3c7814ca505]
aws_instance.myapp-server: Refreshing state... [id=i-0d4a14df8e6221d30]

No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no
differences, so no changes are needed.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

aws_instance_public_ip = "3.28.134.80"
@tehreem-0514 □ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```

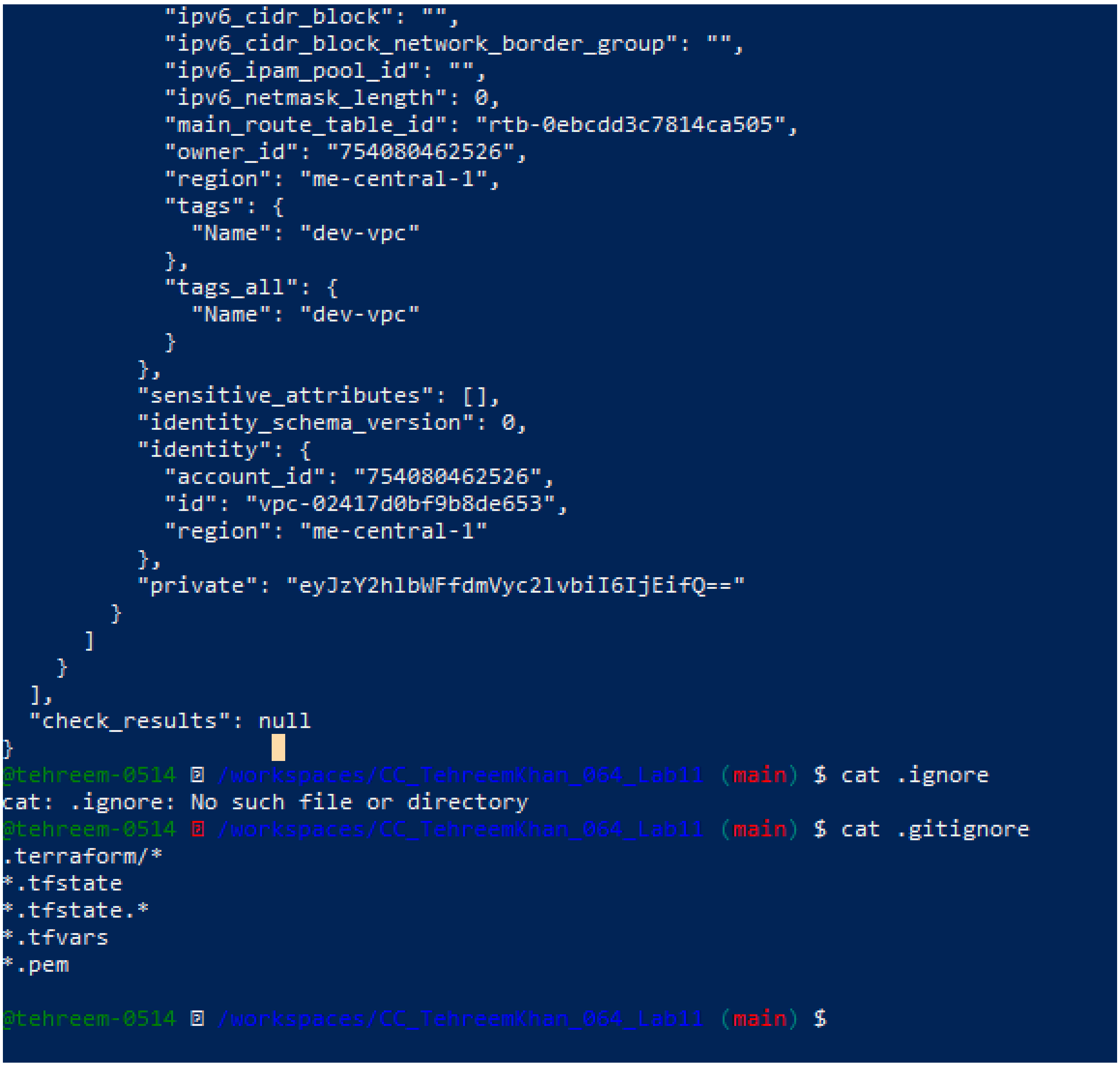
Cleanup:



```
      - instance_tenancy                     = "default" -> null
      - ipv6_netmask_length                  = 0 -> null
      - main_route_table_id                  = "rtb-0ebcdd3c7814ca505" -> null
      - owner_id                             = "754080462526" -> null
      - region                               = "me-central-1" -> null
      - tags                                 = {
          - "Name" = "dev-vpc"
        } -> null
      - tags_all                             = {
          - "Name" = "dev-vpc"
        } -> null
        # (4 unchanged attributes hidden)
    }

Plan: 0 to add, 0 to change, 7 to destroy.

Changes to Outputs:
  - aws_instance_public_ip = "3.28.134.80" -> null
aws_default_route_table.main_rt: Destroying... [id=rtb-0ebcdd3c7814ca505]
aws_key_pair.ssh_key: Destroying... [id=MyED25519Key]
aws_default_route_table.main_rt: Destruction complete after 0s
aws_instance.myapp-server: Destroying... [id=i-0d4a14df8e6221d30]
aws_internet_gateway.myapp_igw: Destroying... [id=igw-08f98ff63110168c7]
aws_key_pair.ssh_key: Destruction complete after 0s
aws_instance.myapp-server: Still destroying... [id=i-0d4a14df8e6221d30, 00m10s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-08f98ff63110168c7, 00m10s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0d4a14df8e6221d30, 00m20s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-08f98ff63110168c7, 00m20s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0d4a14df8e6221d30, 00m30s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-08f98ff63110168c7, 00m30s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0d4a14df8e6221d30, 00m40s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-08f98ff63110168c7, 00m40s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0d4a14df8e6221d30, 00m50s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-08f98ff63110168c7, 00m50s elapsed]
aws_internet_gateway.myapp_igw: Destruction complete after 57s
aws_instance.myapp-server: Still destroying... [id=i-0d4a14df8e6221d30, 01m00s elapsed]
aws_instance.myapp-server: Destruction complete after 1m0s
aws_subnet.myapp_subnet_1: Destroying... [id=subnet-08dd594d4d1f39925]
aws_default_security_group.myapp_sg: Destroying... [id=sg-0318cfdde9863e74b]
aws_default_security_group.myapp_sg: Destruction complete after 0s
aws_subnet.myapp_subnet_1: Destruction complete after 1s
aws_vpc.myapp_vpc: Destroying... [id=vpc-02417d0bf9b8de653]
aws_vpc.myapp_vpc: Destruction complete after 0s

Destroy complete! Resources: 7 destroyed.
@tehreem-0514 ⊠ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```



```
@tehreem-0514 ⊠ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ cat terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 70,
  "lineage": "99fba10e-e73a-69a0-757c-4fa021baa449",
  "outputs": {},
  "resources": [],
  "check_results": null
}
@tehreem-0514 ⊠ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ cat terraform.tfstate
```



```
            "ipv6_cidr_block": "",
            "ipv6_cidr_block_network_border_group": "",
            "ipv6_ipam_pool_id": "",
            "ipv6_netmask_length": 0,
            "main_route_table_id": "rtb-0ebcdd3c7814ca505",
            "owner_id": "754080462526",
            "region": "me-central-1",
            "tags": {
              "Name": "dev-vpc"
            },
            "tags_all": {
              "Name": "dev-vpc"
            }
          },
          "sensitive_attributes": [],
          "identity_schema_version": 0,
          "identity": {
            "account_id": "754080462526",
            "id": "vpc-02417d0bf9b8de653",
            "region": "me-central-1"
          },
          "private": "eyJzY2hlbWFfdmVyc2lvbiI6IjEifQ=="
        }
      ]
    }
  ],
  "check_results": null
}
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ cat .ignore
cat: .ignore: No such file or directory
@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $ cat .gitignore
.terraform/*
*.tfstate
*.tfstate.*
*.tfvars
*.pem

@tehreem-0514 ⍰ /workspaces/CC_TehreemKhan_064_Lab11 (main) $
```