DEPARTMENT OF SOFTWARE ENGINEERING

LAB#11

SUBMITTED TO:

ENGR. MUHAMMAD SHOAIB

ENGR. WAQAS SALEEM

SUBMITTED BY: NEHA AMJAD

REG NO: 2021-BSE-024

```
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ terraform apply -auto-approve
var.subnet_cidr_block
  Enter a value: 10.0.0.0/24



Changes to Outputs:
  + subnet_cidr_block_output = "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"
```

Lab11 > main.tf

```
1  provider "aws" {
2    shared_config_files      = ["~/.aws/config"]
3    shared_credentials_files = ["~/.aws/credentials"]
4  }
5
6  variable "subnet_cidr_block" {
7    type    = string
8    default = "10.0.0.0/24"
9  }
10
11
12 output "subnet_cidr_block_output" {
13   value = var.subnet_cidr_block
14 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS bash - Lab11

```
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ terraform apply -auto-approve
subnet_cidr_block_output = "10.0.0.0/24"
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ terraform apply -auto-approve

No changes. Your infrastructure matches the configuration.

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

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

Outputs:

subnet_cidr_block_output = "10.0.0.0/24"
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $
```

```
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ export TF_VAR_subnet_cidr_block=10.0.20.0/24
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"
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $
```

Lab11 > main.tf

```
1  provider "aws" {
2    shared_config_files      = ["~/.aws/config"]
3    shared_credentials_files = ["~/.aws/credentials"]
4  }
5
```

```
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ printenv | grep TF_VAR_
unset TF_VAR_subnet_cidr_block
printenv | grep TF_VAR_
TF_VAR_subnet_cidr_block=10.0.20.0/24
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $
```

```
is 1.14.3. You can update by downloading from https://www.terraform.io/downloads.html
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (Lab11) $ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v6.28.0...
- Installed hashicorp/aws v6.28.0 (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.
```

[Preview] README.md | main.tf U | terraform.tfvars U

Lab11 > terraform.tfvars

```
1  subnet_cidr_block = "10.0.30.0/24"
2
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS bash - Lab11

```
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ export TF_VAR_subnet_cidr_block=10.0.20.0/24
terraform apply -auto-approve
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ touch terraform.tfvars
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/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"
```

```
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (Lab11) $ touch main.tf
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (Lab11) $ terraform init
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ terraform apply -auto-approve -var "subnet_cidr_b
lock=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"
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $
```

[Preview] README.md | main.tf U

Lab11 > main.tf

```hcl
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
}

```

```
@neha-121 ➜/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ terraform apply -auto-approve -var "api_session_t
oken=my_API_session_Token_12345"                                    terraform apply -auto-approve -var "api_session_t
oken=my_API_session_Token_12345"
  ~ subnet_cidr_block_output = "10.0.40.0/24" -> "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:

api_session_token_output = <sensitive>
subnet_cidr_block_output = "10.0.30.0/24"
@neha-121 ➜/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  ~ api_session_token_output = (sensitive value)

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 = "10.0.30.0/24"
@neha-121 ➜/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ terraform apply -auto-approve

Error: Ephemeral value not allowed

  on main.tf line 35, in output "api_session_token_output":
  35:   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.
```

```
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 let
  }
}

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

```
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ cat terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 5,
  "lineage": "17efa503-abb8-d790-c129-34886e7b6602",
  "outputs": {
    "api_session_token_output": {
      "value": "my_API_session_Token_12345",
      "type": "string",
      "sensitive": true
```

```
@neha-121 →/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ terraform apply -auto-approve -var "subnet_cidr_b
lock=10.0.0"

Error: Invalid value for variable

  on main.tf line 6:
   6: 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:14,3-13.
```

[Preview] README.md | main.tf U | terraform.tfvars U

Lab11 > main.tf

```
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
}

--output text
```

```
@neha-121 ➜/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ aws configure
AWS Access Key ID [None]: AKIAYGZVAOQUXB7AFA4K
AWS Secret Access Key [None]: 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
Default region name [None]: ap-east-1
Default output format [None]: json
@neha-121 ➜/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $
```

```
@neha-121 ➜/workspaces/cc_-nehaamjad-_-2021-BSE-024-/Lab11 (main) $ terraform init
terraform apply -auto-approve

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>
environment = "dev"
monitoring_enabled = true
primary_subnet_id = "subnet-00b47a6f034c9a932"
project_name = "lab_work"
server_config = {
  "backup_enabled" = false
  "instance_type" = "t3.micro"
  "monitoring" = true
  "name" = "web-server"
  "storage_gb" = 20
}
subnet_cidr_block_output = "10.0.0.0/24"
subnet_count = 3
tags = tomap({
  "Environment" = "dev"
  "Owner" = "platform-team"
  "Project" = "sample-app"
})
```

b11 > terraform.tfvars

```
subnet_cidr_block = "10.0.30.0/24"
environment       = "dev"
project_name      = "lab_work"
primary_subnet_id = "subnet-00b47a6f034c9a932"
subnet_count      = 3
monitoring        = true
```

```
variable "subnet_cidr_block" {
  type        = string
  default     = ""
  description = "CIDR block to assign to the application subnet"
  sensitive   = false
  nullable    = false
  ephemeral   = false
}
variable "environment" {}
variable "project_name" {}
variable "primary_subnet_id" {}
variable "subnet_count" {}
variable "monitoring" {}
```