CLOUD COMPUTING LAB

BSE (V-B)

LAB 11

Submitted By: **Urwa Zahra**

Roll No: **2023-BSE-068**

Submitted To: **Sir Muhammad Shoaib**

Task 0 Lab Setup (Codespace & GH CLI)

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ gh codespace list
NAME                                 DISPLAY NAME      REPOSITORY                     BRANCH  STATE      CREATED AT
studious-guide-r45rj7gx6699f56wg     studious guide    Urwa012/cc-urwazahra-2023-BSE...  main*   Available  about 1 day ago
turbo-space-yodel-q759wvqx4gj...     turbo space yodel Urwa012/cc-urwazahra-2023-BSE...  main    Shutdown   about 1 day ago
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ gh codespace ssh -c studious-guide-r45rj7gx6699f56wg
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.
```

Task 1 — Provider & Basic variable (variable precedence)

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ touch main.tf
```

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

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim main.tf
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v6.27.0

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
```

```
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
}
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim main.tf
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply -auto-approve
var.subnet_cidr_block
  Enter a value: yes


Changes to Outputs:
  + subnet_cidr_block_output = "yes"

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

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

Outputs:

subnet_cidr_block_output = "yes"
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim main.tf
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply -auto-approve

Changes to Outputs:
  ~ subnet_cidr_block_output = "yes" -> "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"
```

```
subnet_cidr_block_output = "10.0.0.0/24"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ export TF_VAR_subnet_cidr_block=10.0.20.0/24
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (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 rea
infrastructure.

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

Outputs:

subnet_cidr_block_output = "10.0.20.0/24"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ touch terraform.tfvars
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim terraform.tfvars
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (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"
```

```
subnet_cidr_block_output = "10.0.30.0/24"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply -auto-approve -var "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"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ printenv | grep TF_VAR_
TF_VAR_subnet_cidr_block=10.0.20.0/24
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ unset TF_VAR_subnet_cidr_block
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ printenv | grep TF_VAR_
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

Task 2 — Variable validation & sensitive / ephemeral variables

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

}

output "subnet_cidr_block_output" {
  value = var.subnet_cidr_block
}

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 10.0.0.0/24."
  }
}
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply -auto-approve -var "subnet_cidr_block=10.0.0"

Error: Invalid value for variable

  on main.tf line 12:
  12: 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:20,3-13.

@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
}

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 letters, numbers, hyphens,
underscores."
  }
}

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

```
commands will detect it and remind you to do so if necessary.
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (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 no changes are
needed.

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

Outputs:

api_session_token_output = <sensitive>
subnet_cidr_block_output = "10.0.30.0/24"
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply -auto-approve

Error: Ephemeral value not allowed

  on main.tf line 41, in output "api_session_token_output":
  41:   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.
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (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 no change
needed.

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

Outputs:

api_session_token_output = <sensitive>
subnet_cidr_block_output = "10.0.30.0/24"
```

Task 3 — Project-level variables, locals, and outputs

```
  }
}

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

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

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim main.tf
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ aws ec2 describe-subnets \
  --filters "Name=availability-zone,Values=me-central-1a" \
  --query "Subnets[].SubnetId" \
  --output text
subnet-058d5e01856a8fe1a
```

```
subnet_cidr_block = "10.0.30.0/24"
api_session_token = "my_API_session_Token"


environment = "dev"
project_name = "lab_work"
primary_subnet_id = "<subnet-id-of-me-central-1a>
subnet_count = 3
monitoring = true
~
~
```

```
 GNU nano 7.2                                    locals.tf
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
}
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim main.tf
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply -auto-approve

Changes to Outputs:
  + is_production           = false
  + monitoring_enabled      = true
  + primary_public_subnet   = "<subnet-id-of-me-central-1a>"
  + resource_name           = "lab_work-dev"
  + 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-id-of-me-central-1a>"
resource_name = "lab_work-dev"
subnet_cidr_block_output = "10.0.30.0/24"
subnet_count = 3
```

Task 4 — Maps and Objects

```
    value = local.monitoring_enabled
  }
  variable "tags" {
    type = map(string)
  }

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

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (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-id-of-me-central-1a>"
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"
})
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim main.tf
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim terraform.tfvars
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (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
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-id-of-me-central-1a>"
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"
})
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

Task 5 — Collections: list, tuple, set & mutation via locals

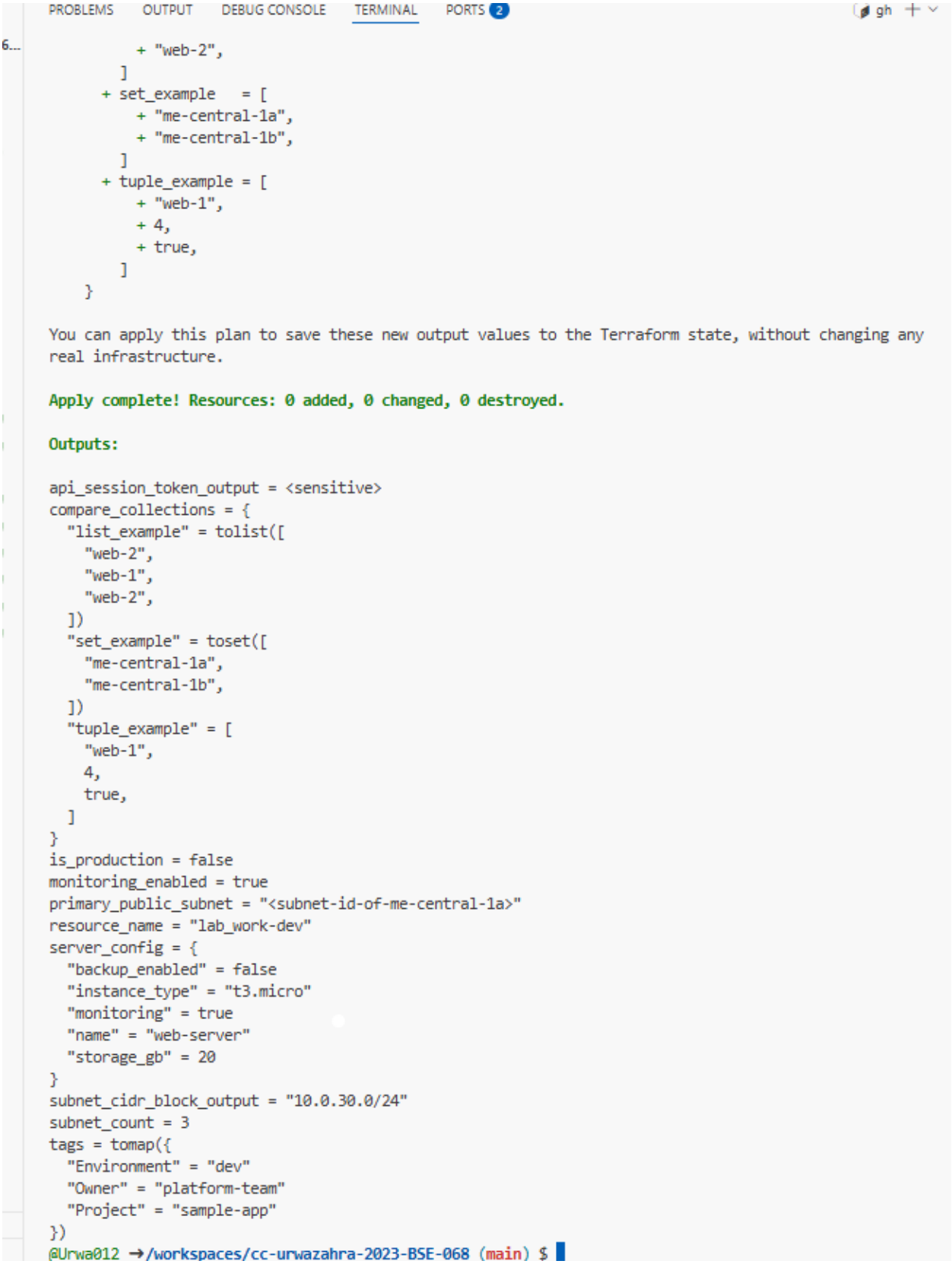
```
output "server_config" {
  value = var.server_config
}

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
  }
}
-- INSERT --
```



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS 2                    gh

6...
          + "web-2",
        ]
      + set_example   = [
          + "me-central-1a",
          + "me-central-1b",
        ]
      + tuple_example = [
          + "web-1",
          + 4,
          + true,
        ]
    }

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-id-of-me-central-1a>"
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"
})
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
 GNU nano 7.2                                                    locals.tf
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

  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"])
}
```

```
      "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-id-of-me-central-1a>"
  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"
  })
  @Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

Task 6 — Null, any type & dynamic values

```
output "mutation_comparison" {
  value = {
    original_tuple = var.server_metadata
    mutated_tuple  = local.mutated_tuple
  }
}
variable "optional_tag" {
  type        = string
  description = "A tag that may or may not be provided"
  default     = null
}
-- INSERT --
```

```
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

  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"])

server_tags = merge(
    { Name = "web-server" },
    var.optional_tag != null ? { Custom = var.optional_tag } : {}
  )
}

~
~
~
~
~
~
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS 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,
  ]
}
optional_tag = {
  "Name" = "web-server"
}
primary_public_subnet = "<subnet-id-of-me-central-1a>"
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"
})
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

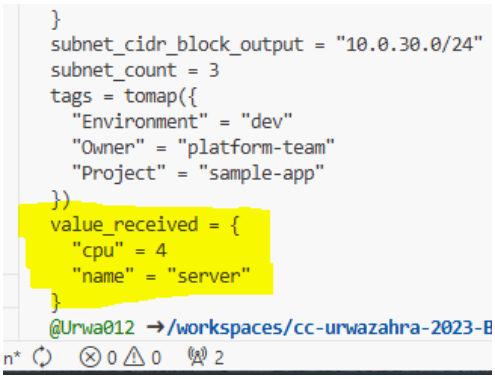
```
}
optional_tag = {
  "Custom" = "dev"
  "Name" = "web-server"
}
primary_public_subnet = "<subnet-id-of-me-central-1a>"
resource_name = "lab_work-dev"
```

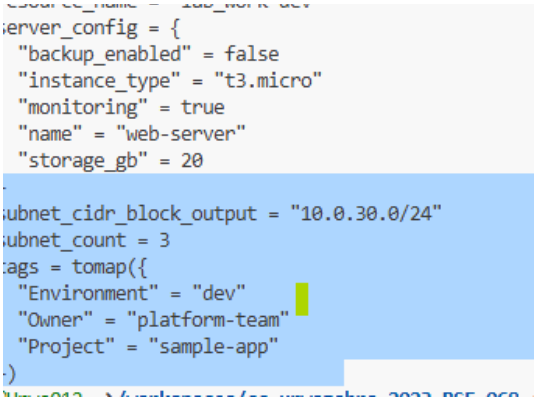
```
    "storage_gb" = 20
  }
  subnet_cidr_block_output = "10.0.30.0/24"
  subnet_count = 3
  tags = tomap({
    "Environment" = "dev"
    "Owner" = "platform-team"
    "Project" = "sample-app"
  })
  value_received = "42"
  @Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
  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"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

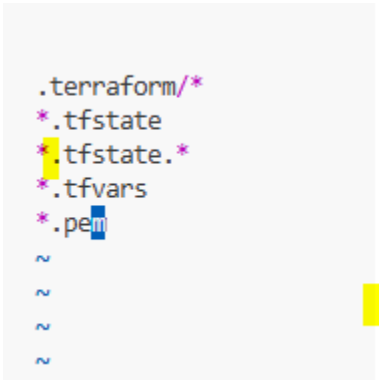
⊗ 0 ⚠ 0 2

```
}
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",
]
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (ma
```





Task 7 — Git ignore



Task 8 — Clean-up then build real infra (VPC, Subnet, IGW, routing, default route table)

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim main.tf
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ rm  terraforms.tf
rm: cannot remove 'terraforms.tf': No such file or directory
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ rm locals.tf
rm: cannot remove 'locals.tf': No such file or directory
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
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" {}
~
~
~
```

```
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"
  }
}
~
~
~
```

```
vpc_cidr_block     = "10.0.0.0/16"
subnet_cidr_block  = "10.0.10.0/24"
availability_zone  = "me-central-1a"
env_prefix         = "dev"
~
~
~
~
```

```
          - "web-2",
          - "web-1",
          - "web-2",
        ]
      - set_example   = [
          - "me-central-1a",
          - "me-central-1b",
        ]
      - tuple_example = [
          - "web-1",
          - 4,
          - true,
        ]
    } -> null
  - is_production            = false -> null
  - monitoring_enabled       = true -> null
  - mutation_comparison      = {
      - mutated_tuple  = [
          - "4",
          - "true",
          - "web-1",
          - "web-2",
        ]
      - original_tuple = [
          - "web-1",
          - 4,
          - true,
        ]
    } -> null
  - optional_tag             = {
      - Custom = "dev"
      - Name   = "web-server"
    } -> null
  - primary_public_subnet    = "<subnet-id-of-me-central-1a>" -> null
  - resource_name            = "lab_work-dev" -> null
  - server_config            = {
      - backup_enabled = false
      - instance_type  = "t3.micro"
      - monitoring     = true
      - name           = "web-server"
      - storage_gb     = 20
    } -> null
  - subnet_cidr_block_output = "10.0.30.0/24" -> null
  - subnet_count             = 3 -> null
  - tags                     = {
      - Environment = "dev"
      - Owner       = "platform-team"
      - Project     = "sample-app"
    } -> null
aws_vpc.myapp_vpc: Creating...
aws_vpc.myapp_vpc: Creation complete after 3s [id=vpc-096ca6ed66e1d01f5]
aws_subnet.myapp_subnet_1: Creating...
aws_subnet.myapp_subnet_1: Creation complete after 1s [id=subnet-0963df7b9e51eb46f]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```

```
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"
  }
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply -auto-approve
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-096ca6ed66e1d01f5]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-0963df7b9e51eb46f]

Terraform used the selected providers to generate the following execution plan. Resource actions
the following symbols:
  + create

Terraform will perform the following actions:

  # aws_internet_gateway.myapp_igw will be created
  + resource "aws_internet_gateway" "myapp_igw" {
      + arn      = (known after apply)
      + id       = (known after apply)
      + owner_id = (known after apply)
      + region   = "me-central-1"
      + tags     = {
          + "Name" = "dev-igw"
        }
      + tags_all = {
          + "Name" = "dev-igw"
        }
      + vpc_id   = "vpc-096ca6ed66e1d01f5"
    }

  # aws_route_table.myapp_route_table will be created
  + resource "aws_route_table" "myapp_route_table" {
      + arn              = (known after apply)
      + id               = (known after apply)
      + owner_id         = (known after apply)
      + propagating_vgws = (known after apply)
      + region           = "me-central-1"
      + route            = [
          + {
              + cidr_block                = "0.0.0.0/0"
              + gateway_id                = (known after apply)
                # (11 unchanged attributes hidden)
            },
        ]
      + tags             = {
          + "Name" = "dev-rt"
        }
      + tags_all         = {
          + "Name" = "dev-rt"
        }
      + vpc_id           = "vpc-096ca6ed66e1d01f5"
    }

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-042b7666dbeb7580a]
aws_route_table.myapp_route_table: Creating...
aws_route_table.myapp_route_table: Creation complete after 1s [id=rtb-02691aa43eaad0451]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply -auto-approve
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-096ca6ed66e1d01f5]
aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-042b7666dbeb7580a]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-0963df7b9e51eb46f]
aws_route_table.myapp_route_table: Refreshing state... [id=rtb-02691aa43eaad0451]

Terraform used the selected providers to generate the following execution plan. Resource actions are
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-02691aa43eaad0451"
      + subnet_id      = "subnet-0963df7b9e51eb46f"
    }

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 1s [id=rtbassoc-0578eb54f85f2b771]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
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"
  }
}
~
~
~
~
~
~
```

```
            # (10 unchanged attributes hidden)
          },
      ]
    + tags                    = {
        + "Name" = "dev-rt"
      }
    + tags_all                = {
        + "Name" = "dev-rt"
      }
    + vpc_id                  = (known after apply)
  }

# 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-1:443915509636:route-table/rtb-02691aa43eaad0451"
    - id               = "rtb-02691aa43eaad0451" -> null
    - owner_id         = "443915509636" -> null
    - propagating_vgws = [] -> null
    - region           = "me-central-1" -> null
    - route            = [
        - {
            - cidr_block                = "0.0.0.0/0"
            - gateway_id                = "igw-042b7666dbeb7580a"
              # (11 unchanged attributes hidden)
          },
      ] -> null
    - tags             = {
        - "Name" = "dev-rt"
      } -> null
    - tags_all         = {
        - "Name" = "dev-rt"
      } -> null
    - vpc_id           = "vpc-096ca6ed66e1d01f5" -> 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-0578eb54f85f2b771" -> null
    - region         = "me-central-1" -> null
    - route_table_id = "rtb-02691aa43eaad0451" -> null
    - subnet_id      = "subnet-0963df7b9e51eb46f" -> 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-0578eb54f85f2b771]
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-02691aa43eaad0451]
aws_route_table.myapp_route_table: Destruction complete after 1s
aws_default_route_table.main_rt: Creation complete after 1s [id=rtb-043baf83ff1f182b1]

Apply complete! Resources: 1 added, 0 changed, 2 destroyed.
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

Task 9 — Security Group, Key Pair, EC2 Instance, user_data & nginx

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

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

variable "my_ip" {}
~
~
```

```
vpc_cidr_block     = "10.0.0.0/16"
subnet_cidr_block  = "10.0.10.0/24"
availability_zone  = "me-central-1a"
env_prefix         = "dev"
4.240.18.
my_ip = "4.240.18.225/32"
instance_type = "t3.micro"
availability_zone = "me-central-1a"   # c
env_prefix = "dev"
~
```

```
}

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"
  }
}

variable "my_ip" {}

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"]
    prefix_list_ids = []
  }

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

```
      + 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-096ca6ed66e1d01f5"
    }

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-0c7226c666eaf7ee8]

  Warning: Value for undeclared variable

  The root module does not declare a variable named "instance_type" but a value was found
  you meant to use this value, add a "variable" block to the configuration.

  To silence these warnings, use TF_VAR_... environment variables to provide certain "glob
  configurations in your organization. To reduce the verbosity of these warnings, use the

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

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ chmod 600 MyED25519Key.pem
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ aws ec2 create-key-pair \
  --key-name MyED25519Key \
  --key-type ed25519 \
  --key-format pem \
  --query 'KeyMaterial' \
  --output text > MyED25519Key.pem
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
  --output text > MyED25519Key.pem
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat .gitignore
.terraform/*
*.tfstate
*.tfstate.*
*.tfvars
*.pem
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
}

resource "aws_instance" "myapp-server" {
  ami                         = "ami-05524d6658fcf35b6" # Amazon Linux 2023
  instance_type               = var.instance_type
  subnet_id                   = aws_subnet.myapp_subnet_1.id
  security_groups             = [aws_default_security_group.default_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
}
-- INSERT --
```

```
    + security_groups                      = [
        + "sg-0c7226c666eaf7ee8",
      ]
    + source_dest_check                    = true
    + spot_instance_request_id             = (known after apply)
    + subnet_id                            = "subnet-0963df7b9e51eb46f"
    + tags                                 = {
        + "Name" = "dev-ec2-instance"
      }
    + tags_all                             = {
        + "Name" = "dev-ec2-instance"
      }
    + tenancy                              = (known after apply)
    + 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)

    + network_interface (known after apply)

    + primary_network_interface (known after apply)

    + private_dns_name_options (known after apply)

    + root_block_device (known after apply)
  }

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 12s [id=i-09d0ff614e8d7e802]

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

Outputs:

aws_instance_public_ip = "51.112.44.228"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
aws_instance_public_ip = "51.112.44.228"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ ssh -i MyED25519Key.pem ec2-user@51.112.44.228
The authenticity of host '51.112.44.228 (51.112.44.228)' can't be established.
ED25519 key fingerprint is SHA256:xFCnbaO5pEYYv1ceJ+S1J6yAGdCWgjF1/Voe4gawAVE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '51.112.44.228' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-10-231 ~]$
```

```
Connection to 51.112.44.228 closed.
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ ssh-keygen -t ed25519 -f ~/.ssh/id_ed25519 -N ""
Generating public/private ed25519 key pair.
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:UOgV1cE3KvyNhGrZ0fYEggYJordf4Vxn51wZ2H9PnSM codespace@codespaces-ea8e9d
The key's randomart image is:
+--[ED25519 256]--+
|. ...o oo+oo o.  |
|..  . +.o + + .o |
|. .  ooo.+o+..o.o|
| . . o.+=o=+.E o=|
|  .   ++S* =o ..+|
|   . .+ . o o   .|
|    ..           |
|                 |
|                 |
+----[SHA256]-----+
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
resource "aws_instance" "myapp-server" {
  ami                         = "ami-05524d6658fcf35b6" # Amazon Linux 2023
  instance_type               = var.instance_type
  subnet_id                   = aws_subnet.myapp_subnet_1.id
  security_groups             = [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
```

```
          - hostname_type                        = "ip-name" -> null
        }

      ~ root_block_device (known after apply)
      - root_block_device {
          - delete_on_termination = true -> null
          - device_name           = "/dev/xvda" -> null
          - encrypted             = false -> null
          - iops                  = 3000 -> null
          - tags                  = {} -> null
          - tags_all              = {} -> null
          - throughput            = 125 -> null
          - volume_id             = "vol-0213c5cdfc63f4944" -> null
          - volume_size           = 8 -> null
          - volume_type           = "gp3" -> null
            # (1 unchanged attribute hidden)
        }
    }

  # aws_key_pair.ssh_key will be created
  + resource "aws_key_pair" "ssh_key" {
      + arn             = (known after apply)
      + fingerprint     = (known after apply)
      + id              = (known after apply)
      + key_name        = "serverkey"
      + key_name_prefix = (known after apply)
      + key_pair_id     = (known after apply)
      + key_type        = (known after apply)
      + public_key      = "ssh-ed25519 AAAAC3NzaC11ZDI1NTE5AAAAIJVCePcIvPHDlzAn/nAdgY/TgbNw
espaces-ea8e9d"
      + region          = "me-central-1"
      + tags_all        = (known after apply)
    }

Plan: 2 to add, 0 to change, 1 to destroy.

Changes to Outputs:
  ~ aws_instance_public_ip = "51.112.44.228" -> (known after apply)
aws_instance.myapp-server: Destroying... [id=i-09d0ff614e8d7e802]
aws_instance.myapp-server: Still destroying... [id=i-09d0ff614e8d7e802, 00m10s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-09d0ff614e8d7e802, 00m20s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-09d0ff614e8d7e802, 00m30s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-09d0ff614e8d7e802, 00m40s elapsed]
aws_instance.myapp-server: Destruction complete after 40s
aws_key_pair.ssh_key: Creating...
aws_key_pair.ssh_key: Creation complete after 0s [id=serverkey]
aws_instance.myapp-server: Creating...
aws_instance.myapp-server: Still creating... [00m10s elapsed]
aws_instance.myapp-server: Creation complete after 13s [id=i-01d4f0024c4dac575]

Apply complete! Resources: 2 added, 0 changed, 1 destroyed.

Outputs:

aws_instance_public_ip = "158.252.34.117"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
aws_instance_public_ip = "158.252.34.117"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ ssh ec2-user@158.252.34.117
The authenticity of host '158.252.34.117 (158.252.34.117)' can't be established.
ED25519 key fingerprint is SHA256:UCeHsyPztfxaGcQXg10Q8QIN4T2gTQpcPXweNXIfdL8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '158.252.34.117' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-10-64 ~]$
```

```
aws_instance_public_ip = "3.28.132.78"
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ curl localhost
curl: (7) Failed to connect to localhost port 80 after 0 ms: Couldn't connect to ser
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ ssh ec2-user@3.28.132.78
The authenticity of host '3.28.132.78 (3.28.132.78)' can't be established.
ED25519 key fingerprint is SHA256:mONNv5nz2DB7jVB5YAg1FtDvo3hTzYAbCrwvVhJMVaw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.28.132.78' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-10-146 ~]$ curl localhost
<!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-10-0-10-146 ~]$
```

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.

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat > entry-script.sh <<'EOF'
#!/bin/bash
yum update -y
yum install -y nginx
systemctl start nginx
systemctl enable nginx
EOF
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

Cleanup:

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

Changes to Outputs:
  - aws_instance_public_ip = "3.28.136.63" -> null
aws_default_route_table.main_rt: Destroying... [id=rtb-043baf83ff1f182b1]
aws_default_route_table.main_rt: Destruction complete after 0s
aws_instance.myapp-server: Destroying... [id=i-02d2f65aa69b29213]
aws_internet_gateway.myapp_igw: Destroying... [id=igw-042b7666dbeb7580a]
aws_instance.myapp-server: Still destroying... [id=i-02d2f65aa69b29213, 00m10s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-042b7666dbeb7580a, 00m10s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-02d2f65aa69b29213, 00m20s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-042b7666dbeb7580a, 00m20s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-02d2f65aa69b29213, 00m30s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-042b7666dbeb7580a, 00m30s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-02d2f65aa69b29213, 00m40s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-042b7666dbeb7580a, 00m40s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-02d2f65aa69b29213, 00m50s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-042b7666dbeb7580a, 00m50s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-02d2f65aa69b29213, 01m00s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-042b7666dbeb7580a, 01m00s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-02d2f65aa69b29213, 01m10s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-042b7666dbeb7580a, 01m10s elapsed]
aws_internet_gateway.myapp_igw: Destruction complete after 1m18s
aws_instance.myapp-server: Still destroying... [id=i-02d2f65aa69b29213, 01m20s elapsed]
aws_instance.myapp-server: Destruction complete after 1m22s
aws_key_pair.ssh_key: Destroying... [id=serverkey]
aws_default_security_group.myapp_sg: Destroying... [id=sg-0c7226c666eaf7ee8]
aws_subnet.myapp_subnet_1: Destroying... [id=subnet-0963df7b9e51eb46f]
aws_default_security_group.myapp_sg: Destruction complete after 0s
aws_key_pair.ssh_key: Destruction complete after 0s
aws_subnet.myapp_subnet_1: Destruction complete after 0s
aws_vpc.myapp_vpc: Destroying... [id=vpc-096ca6ed66e1d01f5]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 7 destroyed.
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 94,
  "lineage": "85cb39e1-fff0-6e4e-f539-23849edb92c4",
  "outputs": {},
  "resources": [],
  "check_results": null
}
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat terraform.tfstate.backup
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 86,
  "lineage": "85cb39e1-fff0-6e4e-f539-23849edb92c4",
  "outputs": {
    "aws_instance_public_ip": {
      "value": "3.28.136.63",
      "type": "string"
    }
  },
  "resources": [
    {
      "mode": "managed",
      "type": "aws_default_route_table",
      "name": "main_rt",
      "provider": "provider[\"registry.terraform.io/hashicorp/aws\"]",
      "instances": [
        {
          "schema_version": 0,
          "attributes": {
            "arn": "arn:aws:ec2:me-central-1:443915509636:route-table/rtb-043baf83ff1f182b1",
            "default_route_table_id": "rtb-043baf83ff1f182b1",
            "id": "rtb-043baf83ff1f182b1",
            "owner_id": "443915509636",
            "propagating_vgws": [],
            "region": "me-central-1",
            "route": [
              {
                "cidr_block": "0.0.0.0/0",
                "core_network_arn": "",
                "destination_prefix_list_id": "",
                "egress_only_gateway_id": "",
                "gateway_id": "igw-042b7666dbeb7580a",
                "instance_id": "",
                "ipv6_cidr_block": "",
                "nat_gateway_id": "",
                "network_interface_id": "",
                "transit_gateway_id": "",
                "vpc_endpoint_id": "",
                "vpc_peering_connection_id": ""
              }
            ],
            "tags": {
```


@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $ git status
On branch main
Your branch is up to date with 'origin/main'.
Untracked files:
(use "git add <file>..." to include in what will be committed)
.gitignore
.terraform.lock.hcl
aws/
awscliv2.zip
entry-script.sh
main.tf
nothing added to commit but untracked files present (use "git add" to track)
@Urwa012 ➜/workspaces/cc-urwazahra-2023-BSE-068 (main) $
0 0 4