CLOUD COMPUTING LAB

BSE (V-B)

LAB 10

Submitted By: Urwa Zahra

Roll No: 2023-BSE-068

Submitted To: Sir Muhammad Shoaib

Task 1: GH CLI installation, authentication, Codespaces (in Codespace)

```
PS C:\Users\TT> winget install --id GitHub.cli
Found an existing package already installed. Trying to upgrade the installed package...
No available upgrade found.
No newer package versions are available from the configured sources.
PS C:\Users\TT> gh auth login -s codespace
```

```
PS C:\Users\TT> gh auth login -s codespace
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Paste an authentication token
Tip: you can generate a Personal Access Token here https://github.com/settings/tokens
The minimum required scopes are 'repo', 'read:org', 'workflow'.
? Paste your authentication token: **************************************
- gh config set -h github.com git_protocol https
✓ Configured git protocol
✓ Logged in as Urwa012
! You were already logged in to this account
```

```
! You were already logged in to this account
PS C:\Users\TT> gh codespace list
error getting codespaces: HTTP 403: Must have admin rights to Repository. (https://api.github.com/user/codes
paces?per_page=30)
```

Task 2: AWS CLI install/configure, Terraform install & verify, create main.tf provider block

```
    inflating: aws/dist/awscli/customizations/sso/index.html
    inflating: aws/dist/awscli/data/metadata.json
    inflating: aws/dist/awscli/data/cli.json
    inflating: aws/dist/awscli/data/ac.index
    inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/top_level.txt
    inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/METADATA
    inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/WHEEL
    inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/INSTALLER
    inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/RECORD
    inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/licenses/AUTHORS.rst
    inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/licenses/LICENSE
    inflating: aws/dist/wheel-0.45.1.dist-info/WHEEL
    inflating: aws/dist/wheel-0.45.1.dist-info/LICENSE.txt
    inflating: aws/dist/wheel-0.45.1.dist-info/METADATA
    inflating: aws/dist/wheel-0.45.1.dist-info/REQUESTED
    inflating: aws/dist/wheel-0.45.1.dist-info/INSTALLER
    inflating: aws/dist/wheel-0.45.1.dist-info/direct_url.json
    inflating: aws/dist/wheel-0.45.1.dist-info/entry_points.txt
    inflating: aws/dist/wheel-0.45.1.dist-info/RECORD
⊗ @Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
  Found preexisting AWS CLI installation: /usr/local/aws-cli/v2/current. Please rerun install script with --update flag.
● @Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
  aws-cli/2.32.28 Python/3.13.11 Linux/6.8.0-1030-azure exe/x86_64.ubuntu.24
○ @Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
output = json
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
[default]
aws_access_key_id = AKIAWOW3...
aws_secret_access_key = QC...k24o5O4UVhGn7BPDvF
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
[default]
region = me-central-1
output = json
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-008 (main) $
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ aws sts get-caller-identity
{
    "UserId": "AIDAWOW3ND6CPIGJXS7MS",
    "Account": "443915509636",
    "Arn": "arn:aws:iam::443915509636:user/urwa"
}
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

B. Install Terraform CLI

1. Install Terraform CLI

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ ^C
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ echo "deb [arch=$(dpkg --print-archite
ture) signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicor
.com $(grep -oP '(?<=UBUNTU_CODENAME=).*' /etc/os-release || lsb_release -cs) main" | sudo tee /
tc/apt/sources.list.d/hashicorp.list
sudo apt update
sudo apt install terraform
which terraform
terraform --version
deb [arch=amd64 signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.release
.hashicorp.com noble main
Hit:1 https://dl.yarnpkg.com/debian stable InRelease
Hit:2 https://apt.releases.hashicorp.com noble InRelease
Hit:3 https://packages.microsoft.com/repos/microsoft-ubuntu-noble-prod noble InRelease
Hit:4 https://repo.anaconda.com/pkgs/misc/debrepo/conda stable InRelease
Hit:5 http://archive.ubuntu.com/ubuntu noble InRelease
Hit:6 http://archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:7 http://archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:8 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
51 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
terraform is already the newest version (1.14.3-1).
0 upgraded, 0 newly installed, 0 to remove and 51 not upgraded.
/usr/bin/terraform
Terraform v1.14.3
on linux_amd64
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

C. Provider Configuration (main.tf)

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

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

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (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.
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat .terraform.lock.hcl
# This file is maintained automatically by "terraform init".
# Manual edits may be lost in future updates.

provider "registry.terraform.io/hashicorp/aws" {
  version = "6.27.0"
  hashes = [
    "h1:bixp2PSsP5ZGBczGCxcbSDn6lF5QFlUXlNroq9cdab4=",
    "zh:177a24b806c72e8484b5cabc93b2b38e3d770ae6f745a998b54d6619fd0e8129",
    "zh:4ac4a85c14fb868a3306b542e6a56c10bd6c6d5a67bc0c9b8f6a9060cf5f3be7",
    "zh:552652185bc85c8ba1da1d65dea47c454728a5c6839c458b6dcd3ce71c19ccfc",
    "zh:60284b8172d09aee91eae0856f09855eaf040ce3a58d6933602ae17c53f8ed04",
    "zh:6be38d156756ca61fb8e7c752cc5d769cd709686700ac4b230f40a6e95b5dbc9",
    "zh:7a409138fae4ef42e3a637e37cb9efedf96459e28a3c764fc4e855e8db9a7485",
    "zh:8070cf5224ed1ed3a3e9a59f7c30ff88bf071c7567165275d477c1738a56c064",
    "zh:894439ef340a9a79f69cd759e27ad11c7826adeca27be1b1ca82b3c9702fa300",
    "zh:89d035eebf08a97c89374ff06040955ddc09f275ecca609d0c9d58d149bef5cf",
    "zh:985b1145d724fc1f38369099e4a5087141885740fd6c0b1dbc492171e73c2e49",
    "zh:9b12af85486a96aedd8d7984b0ff811a4b42e3d88dad1a3fb4c0b580d04fa425",
    "zh:a80b47ae8d1475201c86bd94a5dcb9dd4da5e8b73102a90820b68b66b76d50fd",
    "zh:d3395be1556210f82199b9166a6b2e677cee9c4b67e96e63f6c3a98325ad7ab0",
    "zh:db0b869d09657f6f1e4110b56093c5fcdf9dbdd97c020db1e577b239c0adcbce",
    "zh:ffc72e680370ae7c21f9bd3082c6317730df805c6797427839a6b6b7e9a26a01",
  ]
}
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ ls .terraform/
providers
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

Task 3 — VPC/Subnet Creation, Initialization, Verification

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}
resource "aws_vpc" "development_vpc" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_subnet" "dev_subnet_1" {
  vpc_id            = aws_vpc.development_vpc.id
  cidr_block = "10.0.10.0/24"
  availability_zone = "me-central-1a"
}
~
~
~
~
```

```
                + availability_zone                              = "me-central-1a"
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply
      + cidr_block                                      = "10.0.10.0/24"
      + enable_dns64                                    = false
      + enable_resource_name_dns_a_record_on_launch     = false
      + enable_resource_name_dns_aaaa_record_on_launch  = false
      + id                                              = (known after apply)
      + ipv6_cidr_block_association_id                  = (known after apply)
      + ipv6_native                                     = false
      + map_public_ip_on_launch                         = false
      + owner_id                                        = (known after apply)
      + private_dns_hostname_type_on_launch             = (known after apply)
      + region                                          = "me-central-1"
      + tags_all                                        = (known after apply)
      + vpc_id                                          = (known after apply)
    }

  # aws_vpc.development_vpc will be created
  + resource "aws_vpc" "development_vpc" {
      + arn                                   = (known after apply)
      + cidr_block                            = "10.0.0.0/16"
      + default_network_acl_id                = (known after apply)
      + default_route_table_id                = (known after apply)
      + default_security_group_id             = (known after apply)
      + dhcp_options_id                       = (known after apply)
      + enable_dns_hostnames                  = (known after apply)
      + enable_dns_support                    = true
      + enable_network_address_usage_metrics  = (known after apply)
      + id                                    = (known after apply)
      + instance_tenancy                      = "default"
      + ipv6_association_id                   = (known after apply)
      + ipv6_cidr_block                       = (known after apply)
      + ipv6_cidr_block_network_border_group  = (known after apply)
      + main_route_table_id                   = (known after apply)
      + owner_id                              = (known after apply)
      + region                                = "me-central-1"
      + tags_all                              = (known after apply)
    }

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

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_vpc.development_vpc: Creating...
aws_vpc.development_vpc: Creation complete after 1s [id=vpc-0b0c56944039c42b1]
aws_subnet.dev_subnet_1: Creating...
aws_subnet.dev_subnet_1: Creation complete after 1s [id=subnet-01236e2e570107208]

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

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ aws ec2 describe-subnets --filter "Name=subnet-id,Values=su
bnet-01236e2e570107208"
{
    "Subnets": [
        {
            "AvailabilityZoneId": "mec1-az1",
            "MapCustomerOwnedIpOnLaunch": false,
            "OwnerId": "443915509636",
            "AssignIpv6AddressOnCreation": false,
            "Ipv6CidrBlockAssociationSet": [],
            "SubnetArn": "arn:aws:ec2:me-central-1:443915509636:subnet/subnet-01236e2e570107208",
            "EnableDns64": false,
            "Ipv6Native": false,
            "PrivateDnsNameOptionsOnLaunch": {
                "HostnameType": "ip-name",
                "EnableResourceNameDnsARecord": false,
                "EnableResourceNameDnsAAAARecord": false
            },
            "BlockPublicAccessStates": {
                "InternetGatewayBlockMode": "off"
            },
            "SubnetId": "subnet-01236e2e570107208",
            "State": "available",
            "VpcId": "vpc-0b0c56944039c42b1",
            "CidrBlock": "10.0.10.0/24",
            "AvailableIpAddressCount": 251,
            "AvailabilityZone": "me-central-1a",
            "DefaultForAz": false,
            "MapPublicIpOnLaunch": false
        }
    ]
}

@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ aws ec2 describe-vpcs --filter "Name=vpc-id,Values=vpc-0b
56944039c42b1"
{
    "Vpcs": [
        {
            "OwnerId": "443915509636",
            "InstanceTenancy": "default",
            "CidrBlockAssociationSet": [
                {
                    "AssociationId": "vpc-cidr-assoc-046a0746e6addcfea",
                    "CidrBlock": "10.0.0.0/16",
                    "CidrBlockState": {
                        "State": "associated"
                    }
                }
            ],
            "IsDefault": false,
            "BlockPublicAccessStates": {
                "InternetGatewayBlockMode": "off"
            },
            "VpcId": "vpc-0b0c56944039c42b1",
            "State": "available",
            "CidrBlock": "10.0.0.0/16",
            "DhcpOptionsId": "dopt-0426dabcc5c93c0f9"
        }
    ]
}
```

Task 4 — Data Source, Targeted Destroy, Tags

A. Data Source & Resource Creation

```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}
resource "aws_vpc" "development_vpc" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_subnet" "dev_subnet_1" {
  vpc_id            = aws_vpc.development_vpc.id
  cidr_block        = "10.0.10.0/24"
  availability_zone = "me-central-1a"
}

data "aws_vpc" "existing_vpc" {
  default = true
}

resource "aws_subnet" "dev_subnet_1_existing" {
  vpc_id            = data.aws_vpc.existing_vpc.id
  cidr_block        = "172.31.48.0/24"
  availability_zone = "me-central-1a"
}
~
~
```

```
● @Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply
data.aws_vpc.existing_vpc: Reading...
aws_vpc.development_vpc: Refreshing state... [id=vpc-0b0c56944039c42b1]
data.aws_vpc.existing_vpc: Read complete after 1s [id=vpc-0b7740d2d5f400590]
aws_subnet.dev_subnet_1: Refreshing state... [id=subnet-01236e2e570107208]

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

Terraform will perform the following actions:

  # aws_subnet.dev_subnet_1_existing will be created
  + resource "aws_subnet" "dev_subnet_1_existing" {
      + arn                                            = (known after apply)
      + assign_ipv6_address_on_creation                = false
      + availability_zone                              = "me-central-1a"
      + availability_zone_id                           = (known after apply)
      + cidr_block                                     = "172.31.48.0/24"
      + enable_dns64                                   = false
      + enable_resource_name_dns_a_record_on_launch    = false
      + enable_resource_name_dns_aaaa_record_on_launch = false
      + id                                             = (known after apply)
      + ipv6_cidr_block_association_id                 = (known after apply)
      + ipv6_native                                    = false
      + map_public_ip_on_launch                        = false
      + owner_id                                       = (known after apply)
      + private_dns_hostname_type_on_launch            = (known after apply)
      + region                                         = "me-central-1"
      + tags_all                                       = (known after apply)
      + vpc_id                                         = "vpc-0b7740d2d5f400590"
    }

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

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_subnet.dev_subnet_1_existing: Creating...
aws_subnet.dev_subnet_1_existing: Creation complete after 1s [id=subnet-0bd9a9a62ee624191]

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

B. Targeted Destroy & Refresh

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform destroy -target=aws_subnet.dev_subnet_1_exist
      - enable_lni_at_device_index                   = 0 -> null
      - enable_resource_name_dns_a_record_on_launch   = false -> null
      - enable_resource_name_dns_aaaa_record_on_launch = false -> null
      - id                                            = "subnet-0bd9a9a62ee624191" -> null
      - ipv6_native                                   = false -> null
      - map_customer_owned_ip_on_launch               = false -> null
      - map_public_ip_on_launch                       = false -> null
      - owner_id                                      = "443915509636" -> null
      - private_dns_hostname_type_on_launch           = "ip-name" -> null
      - region                                        = "me-central-1" -> null
      - tags                                          = {} -> null
      - tags_all                                      = {} -> null
      - vpc_id                                        = "vpc-0b7740d2d5f400590" -> null
        # (4 unchanged attributes hidden)
    }

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

 │ Warning: Resource targeting is in effect
 │
 │ You are creating a plan with the -target option, which means that the result of this plan may not represent al
 │ the changes requested by the current configuration.
 │
 │ The -target option is not for routine use, and is provided only for exceptional situations such as recovering
 │ errors or mistakes, or when Terraform specifically suggests to use it as part of an error message.


Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

aws_subnet.dev_subnet_1_existing: Destroying... [id=subnet-0bd9a9a62ee624191]
aws_subnet.dev_subnet_1_existing: Destruction complete after 0s

 │ Warning: Applied changes may be incomplete
 │
 │ The plan was created with the -target option in effect, so some changes requested in the configuration may hav
 │ been ignored and the output values may not be fully updated. Run the following command to verify that no other
 │ changes are pending:
 │     terraform plan
 │
 │ Note that the -target option is not suitable for routine use, and is provided only for exceptional situations
 │ as recovering from errors or mistakes, or when Terraform specifically suggests to use it as part of an error
 │ message.


Destroy complete! Resources: 1 destroyed.
● @Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform refresh
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform refresh
data.aws_vpc.existing_vpc: Reading...
aws_vpc.development_vpc: Refreshing state... [id=vpc-0b0c56944039c42b1]
data.aws_vpc.existing_vpc: Read complete after 1s [id=vpc-0b7740d2d5f400590]
aws_subnet.dev_subnet_1: Refreshing state... [id=subnet-01236e2e570107208]
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply
data.aws_vpc.existing_vpc: Reading...
aws_vpc.development_vpc: Refreshing state... [id=vpc-0b0c56944039c42b1]
data.aws_vpc.existing_vpc: Read complete after 1s [id=vpc-0b7740d2d5f400590]
aws_subnet.dev_subnet_1: Refreshing state... [id=subnet-01236e2e570107208]

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_subnet.dev_subnet_1_existing will be created
  + resource "aws_subnet" "dev_subnet_1_existing" {
      + arn                                            = (known after apply)
      + assign_ipv6_address_on_creation                = false
      + availability_zone                              = "me-central-1a"
      + availability_zone_id                           = (known after apply)
      + cidr_block                                     = "172.31.48.0/24"
      + enable_dns64                                   = false
      + enable_resource_name_dns_a_record_on_launch    = false
      + enable_resource_name_dns_aaaa_record_on_launch = false
      + id                                             = (known after apply)
      + ipv6_cidr_block_association_id                 = (known after apply)
      + ipv6_native                                    = false
      + map_public_ip_on_launch                        = false
      + owner_id                                       = (known after apply)
      + private_dns_hostname_type_on_launch            = (known after apply)
      + region                                         = "me-central-1"
      + tags_all                                       = (known after apply)
      + vpc_id                                         = "vpc-0b7740d2d5f400590"
    }

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

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_subnet.dev_subnet_1_existing: Creating...
aws_subnet.dev_subnet_1_existing: Creation complete after 1s [id=subnet-0926e004a56bd66e1]

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

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform destroy
        - tags_all                                = {} -> null
        - vpc_id                                  = "vpc-0b7740d2d5f400590" ->
          # (4 unchanged attributes hidden)
    }

  # aws_vpc.development_vpc will be destroyed
  - resource "aws_vpc" "development_vpc" {
      - arn                                 = "arn:aws:ec2:me-central-1:44391550963
ull
      - assign_generated_ipv6_cidr_block    = false -> null
      - cidr_block                          = "10.0.0.0/16" -> null
      - default_network_acl_id              = "acl-09e5496d98831fca3" -> null
      - default_route_table_id              = "rtb-02a8eadc90be5fce6" -> null
      - default_security_group_id           = "sg-0b09c49d389fee7fd" -> null
      - dhcp_options_id                     = "dopt-0426dabcc5c93c0f9" -> null
      - enable_dns_hostnames                = false -> null
      - enable_dns_support                  = true -> null
      - enable_network_address_usage_metrics = false -> null
      - id                                  = "vpc-0b0c56944039c42b1" -> null
      - instance_tenancy                    = "default" -> null
      - ipv6_netmask_length                 = 0 -> null
      - main_route_table_id                 = "rtb-02a8eadc90be5fce6" -> null
      - owner_id                            = "443915509636" -> null
      - region                              = "me-central-1" -> null
      - tags                                = {} -> null
      - tags_all                            = {} -> null
          # (4 unchanged attributes hidden)
    }

Plan: 0 to add, 0 to change, 3 to destroy.

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

aws_subnet.dev_subnet_1_existing: Destroying... [id=subnet-0926e004a56bd66e1]
aws_subnet.dev_subnet_1: Destroying... [id=subnet-01236e2e570107208]
aws_subnet.dev_subnet_1_existing: Destruction complete after 1s
aws_subnet.dev_subnet_1: Destruction complete after 1s
aws_vpc.development_vpc: Destroying... [id=vpc-0b0c56944039c42b1]
aws_vpc.development_vpc: Destruction complete after 1s

Destroy complete! Resources: 3 destroyed.
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform plan
        + region                                      = "me-central-1"
        + tags_all                                    = (known after apply)
        + vpc_id                                      = (known after apply)
    }

  # aws_subnet.dev_subnet_1_existing will be created
  + resource "aws_subnet" "dev_subnet_1_existing" {
        + arn                                            = (known after apply)
        + assign_ipv6_address_on_creation                = false
        + availability_zone                              = "me-central-1a"
        + availability_zone_id                           = (known after apply)
        + cidr_block                                     = "172.31.48.0/24"
        + enable_dns64                                   = false
        + enable_resource_name_dns_a_record_on_launch    = false
        + enable_resource_name_dns_aaaa_record_on_launch = false
        + id                                             = (known after apply)
        + ipv6_cidr_block_association_id                 = (known after apply)
        + ipv6_native                                    = false
        + map_public_ip_on_launch                        = false
        + owner_id                                       = (known after apply)
        + private_dns_hostname_type_on_launch            = (known after apply)
        + region                                         = "me-central-1"
        + tags_all                                       = (known after apply)
        + vpc_id                                         = "vpc-0b7740d2d5f400590"
    }

  # aws_vpc.development_vpc will be created
  + resource "aws_vpc" "development_vpc" {
        + arn                                 = (known after apply)
        + cidr_block                          = "10.0.0.0/16"
        + default_network_acl_id              = (known after apply)
        + default_route_table_id              = (known after apply)
        + default_security_group_id           = (known after apply)
        + dhcp_options_id                     = (known after apply)
        + enable_dns_hostnames                = (known after apply)
        + enable_dns_support                  = true
        + enable_network_address_usage_metrics = (known after apply)
        + id                                  = (known after apply)
        + instance_tenancy                    = "default"
        + ipv6_association_id                 = (known after apply)
        + ipv6_cidr_block                     = (known after apply)
        + ipv6_cidr_block_network_border_group = (known after apply)
        + main_route_table_id                 = (known after apply)
        + owner_id                            = (known after apply)
        + region                              = "me-central-1"
        + tags_all                            = (known after apply)
    }

Plan: 3 to add, 0 to change, 0 to destroy.
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply
        + availability_zone_id                          = (known after apply)
        + cidr_block                                    = "172.31.48.0/24"
        + enable_dns64                                  = false
        + enable_resource_name_dns_a_record_on_launch   = false
        + enable_resource_name_dns_aaaa_record_on_launch = false
        + id                                            = (known after apply)
        + ipv6_cidr_block_association_id                = (known after apply)
        + ipv6_native                                   = false
        + map_public_ip_on_launch                       = false
        + owner_id                                      = (known after apply)
        + private_dns_hostname_type_on_launch           = (known after apply)
        + region                                        = "me-central-1"
        + tags_all                                      = (known after apply)
        + vpc_id                                        = "vpc-0b7740d2d5f400590"
    }

  # aws_vpc.development_vpc will be created
  + resource "aws_vpc" "development_vpc" {
        + arn                                  = (known after apply)
        + cidr_block                           = "10.0.0.0/16"
        + default_network_acl_id               = (known after apply)
        + default_route_table_id               = (known after apply)
        + default_security_group_id            = (known after apply)
        + dhcp_options_id                      = (known after apply)
        + enable_dns_hostnames                 = (known after apply)
        + enable_dns_support                   = true
        + enable_network_address_usage_metrics = (known after apply)
        + id                                   = (known after apply)
        + instance_tenancy                     = "default"
        + ipv6_association_id                  = (known after apply)
        + ipv6_cidr_block                      = (known after apply)
        + ipv6_cidr_block_network_border_group = (known after apply)
        + main_route_table_id                  = (known after apply)
        + owner_id                             = (known after apply)
        + region                               = "me-central-1"
        + tags_all                             = (known after apply)
    }

Plan: 3 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_subnet.dev_subnet_1_existing: Creating...
aws_vpc.development_vpc: Creating...
aws_subnet.dev_subnet_1_existing: Creation complete after 1s [id=subnet-04d175744f9c209bf]
aws_vpc.development_vpc: Creation complete after 2s [id=vpc-0c12198ea7f0caa5d]
aws_subnet.dev_subnet_1: Creating...
aws_subnet.dev_subnet_1: Creation complete after 0s [id=subnet-0695a5311c98e1714]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

B. Tagging Resources

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

resource "aws_vpc" "development_vpc" {
  cidr_block = "10.0.0.0/16"
  tags = {
    Name: "development"
    vpc_env = "dev"
  }
}

resource "aws_subnet" "dev_subnet_1" {
  vpc_id      = aws_vpc.development_vpc.id
  cidr_block = "10.0.10.0/24"
  availability_zone = "me-central-1a"
  tags = {
    Name: "subnet-1-dev"
  }
}
data "aws_vpc" "existing_vpc" {
  default = true
}
resource "aws_subnet" "dev_subnet_1_existing" {
  vpc_id      = data.aws_vpc.existing_vpc.id
  cidr_block = "172.31.48.0/24"
  availability_zone = "me-central-1a"
  tags = {
    Name: "subnet-1-default"
  }
}
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform refresh
 data.aws_vpc.existing_vpc: Reading...
 aws_vpc.development_vpc: Refreshing state... [id=vpc-0c12198ea7f0caa5d]
 data.aws_vpc.existing_vpc: Read complete after 1s [id=vpc-0b7740d2d5f400590]
 aws_subnet.dev_subnet_1_existing: Refreshing state... [id=subnet-04d175744f9c209bf]
 aws_subnet.dev_subnet_1: Refreshing state... [id=subnet-0695a5311c98e1714]
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply -auto-approve
```

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

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated w
the following symbols:
  ~ update in-place

Terraform will perform the following actions:

  # aws_subnet.dev_subnet_1 will be updated in-place
  ~ resource "aws_subnet" "dev_subnet_1" {
        id                                    = "subnet-0695a5311c98e1714"
      ~ tags                                  = {
          + "Name" = "subnet-1-dev"
        }
      ~ tags_all                              = {
          + "Name" = "subnet-1-dev"
        }
        # (20 unchanged attributes hidden)
    }

  # aws_subnet.dev_subnet_1_existing will be updated in-place
  ~ resource "aws_subnet" "dev_subnet_1_existing" {
        id                                    = "subnet-04d175744f9c209bf"
      ~ tags                                  = {
          + "Name" = "subnet-1-default"
        }
      ~ tags_all                              = {
          + "Name" = "subnet-1-default"
        }
        # (20 unchanged attributes hidden)
    }

  # aws_vpc.development_vpc will be updated in-place
  ~ resource "aws_vpc" "development_vpc" {
        id                          = "vpc-0c12198ea7f0caa5d"
      ~ tags                        = {
          + "Name"    = "development"
          + "vpc_env" = "dev"
        }
      ~ tags_all                    = {
          + "Name"    = "development"
          + "vpc_env" = "dev"
        }
        # (19 unchanged attributes hidden)
    }

Plan: 0 to add, 3 to change, 0 to destroy.
aws_subnet.dev_subnet_1_existing: Modifying... [id=subnet-04d175744f9c209bf]
aws_vpc.development_vpc: Modifying... [id=vpc-0c12198ea7f0caa5d]
aws_subnet.dev_subnet_1_existing: Modifications complete after 0s [id=subnet-04d175744f9c209bf]
aws_vpc.development_vpc: Modifications complete after 1s [id=vpc-0c12198ea7f0caa5d]
aws_subnet.dev_subnet_1: Modifying... [id=subnet-0695a5311c98e1714]
aws_subnet.dev_subnet_1: Modifications complete after 0s [id=subnet-0695a5311c98e1714]

Apply complete! Resources: 0 added, 3 changed, 0 destroyed.
```

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

resource "aws_vpc" "development_vpc" {
  cidr_block = "10.0.0.0/16"
  tags = {
    Name: "development"
  }
}

resource "aws_subnet" "dev_subnet_1" {
  vpc_id     = aws_vpc.development_vpc.id
  cidr_block = "10.0.10.0/24"
  availability_zone = "me-central-1a"
  tags = {
    Name: "subnet-1-dev"
  }
}
data "aws_vpc" "existing_vpc" {
  default = true
}
resource "aws_subnet" "dev_subnet_1_existing" {
  vpc_id     = data.aws_vpc.existing_vpc.id
  cidr_block = "172.31.48.0/24"
  availability_zone = "me-central-1a"
  tags = {
    Name: "subnet-1-default"
  }
}
~
```

```
Apply complete! Resources: 0 added, 3 changed, 0 destroyed.
▶ @Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ vim main.tf
▶ @Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply
data.aws_vpc.existing_vpc: Reading...
aws_vpc.development_vpc: Refreshing state... [id=vpc-0c12198ea7f0caa5d]
data.aws_vpc.existing_vpc: Read complete after 1s [id=vpc-0b7740d2d5f400590]
aws_subnet.dev_subnet_1_existing: Refreshing state... [id=subnet-04d175744f9c209bf]
aws_subnet.dev_subnet_1: Refreshing state... [id=subnet-0695a5311c98e1714]

Terraform used the selected providers to generate the following execution plan. Resource actions are i
the following symbols:
  ~ update in-place

Terraform will perform the following actions:

  # aws_vpc.development_vpc will be updated in-place
  ~ resource "aws_vpc" "development_vpc" {
        id                                 = "vpc-0c12198ea7f0caa5d"
      ~ tags                               = {
            "Name"    = "development"
          - "vpc_env" = "dev" -> null
        }
      ~ tags_all                           = {
          - "vpc_env" = "dev" -> null
            # (1 unchanged element hidden)
        }
        # (19 unchanged attributes hidden)
    }

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

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_vpc.development_vpc: Modifying... [id=vpc-0c12198ea7f0caa5d]
aws_vpc.development_vpc: Modifications complete after 1s [id=vpc-0c12198ea7f0caa5d]

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

Task 5 — State File Inspection & Terraform State Command

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform destroy
        - tags                                       = {
            - "Name" = "subnet-1-default"
          } -> null
        - tags_all                                   = {
            - "Name" = "subnet-1-default"
          } -> null
        - vpc_id                                     = "vpc-0b7740d2d5f400590" -> null
          # (4 unchanged attributes hidden)
      }

    # aws_vpc.development_vpc will be destroyed
    - resource "aws_vpc" "development_vpc" {
        - arn                                = "arn:aws:ec2:me-central-1:443915509636:vp
ull
        - assign_generated_ipv6_cidr_block   = false -> null
        - cidr_block                         = "10.0.0.0/16" -> null
        - default_network_acl_id             = "acl-03a4e9640a7878761" -> null
        - default_route_table_id             = "rtb-0665a0eb511e79adc" -> null
        - default_security_group_id          = "sg-0d5b3938754d00fcc" -> null
        - dhcp_options_id                    = "dopt-0426dabcc5c93c0f9" -> null
        - enable_dns_hostnames               = false -> null
        - enable_dns_support                 = true -> null
        - enable_network_address_usage_metrics = false -> null
        - id                                 = "vpc-0c12198ea7f0caa5d" -> null
        - instance_tenancy                   = "default" -> null
        - ipv6_netmask_length                = 0 -> null
        - main_route_table_id                = "rtb-0665a0eb511e79adc" -> null
        - owner_id                           = "443915509636" -> null
        - region                             = "me-central-1" -> null
        - tags                               = {
            - "Name" = "development"
          } -> null
        - tags_all                           = {
            - "Name" = "development"
          } -> null
          # (4 unchanged attributes hidden)
      }

Plan: 0 to add, 0 to change, 3 to destroy.

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

aws_subnet.dev_subnet_1_existing: Destroying... [id=subnet-04d175744f9c209bf]
aws_subnet.dev_subnet_1: Destroying... [id=subnet-0695a5311c98e1714]
aws_subnet.dev_subnet_1_existing: Destruction complete after 1s
aws_subnet.dev_subnet_1: Destruction complete after 1s
aws_vpc.development_vpc: Destroying... [id=vpc-0c12198ea7f0caa5d]
aws_vpc.development_vpc: Destruction complete after 1s

Destroy complete! Resources: 3 destroyed.
```

```
Destroy complete! Resources: 3 destroyed.
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 31,
  "lineage": "85cb39e1-fff0-6e4e-f539-23849edb92c4",
  "outputs": {},
  "resources": [],
  "check_results": null
}
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat terraform.tfstate.backup
        ]
      },
      {
        "mode": "managed",
        "type": "aws_vpc",
        "name": "development_vpc",
        "provider": "provider[\"registry.terraform.io/hashicorp/aws\"]",
        "instances": [
          {
            "schema_version": 1,
            "attributes": {
              "arn": "arn:aws:ec2:me-central-1:443915509636:vpc/vpc-0c12198ea7f0caa5d",
              "assign_generated_ipv6_cidr_block": false,
              "cidr_block": "10.0.0.0/16",
              "default_network_acl_id": "acl-03a4e9640a7878761",
              "default_route_table_id": "rtb-0665a0eb511e79adc",
              "default_security_group_id": "sg-0d5b3938754d00fcc",
              "dhcp_options_id": "dopt-0426dabcc5c93c0f9",
              "enable_dns_hostnames": false,
              "enable_dns_support": true,
              "enable_network_address_usage_metrics": false,
              "id": "vpc-0c12198ea7f0caa5d",
              "instance_tenancy": "default",
              "ipv4_ipam_pool_id": null,
              "ipv4_netmask_length": null,
              "ipv6_association_id": "",
              "ipv6_cidr_block": "",
              "ipv6_cidr_block_network_border_group": "",
              "ipv6_ipam_pool_id": "",
              "ipv6_netmask_length": 0,
              "main_route_table_id": "rtb-0665a0eb511e79adc",
              "owner_id": "443915509636",
              "region": "me-central-1",
              "tags": {
                "Name": "development"
              },
              "tags_all": {
                "Name": "development"
              }
            },
            "sensitive_attributes": [],
            "identity_schema_version": 0,
            "identity": {
              "account_id": "443915509636",
              "id": "vpc-0c12198ea7f0caa5d",
              "region": "me-central-1"
            },
            "private": "eyJzY2hlbWFfdmVyc2lvbiI6IjEifQ=="
          }
        ]
      }
    ],
    "check_results": null
}
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform apply
        + private_dns_hostname_type_on_launch      = (known after apply)
        + region                                   = "me-central-1"
        + tags                                     = {
            + "Name" = "subnet-1-default"
          }
        + tags_all                                 = {
            + "Name" = "subnet-1-default"
          }
        + vpc_id                                   = "vpc-0b7740d2d5f400590"
    }

  # aws_vpc.development_vpc will be created
  + resource "aws_vpc" "development_vpc" {
      + arn                                  = (known after apply)
      + cidr_block                           = "10.0.0.0/16"
      + default_network_acl_id               = (known after apply)
      + default_route_table_id               = (known after apply)
      + default_security_group_id            = (known after apply)
      + dhcp_options_id                      = (known after apply)
      + enable_dns_hostnames                 = (known after apply)
      + enable_dns_support                   = true
      + enable_network_address_usage_metrics = (known after apply)
      + id                                   = (known after apply)
      + instance_tenancy                     = "default"
      + ipv6_association_id                  = (known after apply)
      + ipv6_cidr_block                      = (known after apply)
      + ipv6_cidr_block_network_border_group = (known after apply)
      + main_route_table_id                  = (known after apply)
      + owner_id                             = (known after apply)
      + region                               = "me-central-1"
      + tags                                 = {
          + "Name" = "development"
        }
      + tags_all                             = {
          + "Name" = "development"
        }
    }

Plan: 3 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_vpc.development_vpc: Creating...
aws_subnet.dev_subnet_1_existing: Creating...
aws_subnet.dev_subnet_1_existing: Creation complete after 1s [id=subnet-014bb9108f53cfa35]
aws_vpc.development_vpc: Creation complete after 1s [id=vpc-04024e4c880ab1358]
aws_subnet.dev_subnet_1: Creating...
aws_subnet.dev_subnet_1: Creation complete after 1s [id=subnet-01bfeb495dadc4aa9]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat terraform.tfstate
      ]
    },
    {
      "mode": "managed",
      "type": "aws_vpc",
      "name": "development_vpc",
      "provider": "provider[\"registry.terraform.io/hashicorp/aws\"]",
      "instances": [
        {
          "schema_version": 1,
          "attributes": {
            "arn": "arn:aws:ec2:me-central-1:443915509636:vpc/vpc-04024e4c880ab1358
            "assign_generated_ipv6_cidr_block": false,
            "cidr_block": "10.0.0.0/16",
            "default_network_acl_id": "acl-040674ff1394c131a",
            "default_route_table_id": "rtb-0514cac11257ebda6",
            "default_security_group_id": "sg-05386a7e6f34196fc",
            "dhcp_options_id": "dopt-0426dabcc5c93c0f9",
            "enable_dns_hostnames": false,
            "enable_dns_support": true,
            "enable_network_address_usage_metrics": false,
            "id": "vpc-04024e4c880ab1358",
            "instance_tenancy": "default",
            "ipv4_ipam_pool_id": null,
            "ipv4_netmask_length": null,
            "ipv6_association_id": "",
            "ipv6_cidr_block": "",
            "ipv6_cidr_block_network_border_group": "",
            "ipv6_ipam_pool_id": "",
            "ipv6_netmask_length": 0,
            "main_route_table_id": "rtb-0514cac11257ebda6",
            "owner_id": "443915509636",
            "region": "me-central-1",
            "tags": {
              "Name": "development"
            },
            "tags_all": {
              "Name": "development"
            }
          },
          "sensitive_attributes": [],
          "identity_schema_version": 0,
          "identity": {
            "account_id": "443915509636",
            "id": "vpc-04024e4c880ab1358",
            "region": "me-central-1"
          },
          "private": "eyJzY2hlbWFfdmVyc2lvbiI6IjEifQ=="
        }
      ]
    }
  ],
  "check_results": null
}
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat terraform.tfstate.backup
{
    "version": 4,
    "terraform_version": "1.14.3",
    "serial": 31,
    "lineage": "85cb39e1-fff0-6e4e-f539-23849edb92c4",
    "outputs": {},
    "resources": [],
    "check_results": null
}
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform state list
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform state list
data.aws_vpc.existing_vpc
aws_subnet.dev_subnet_1
aws_subnet.dev_subnet_1_existing
aws_vpc.development_vpc
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

Task 6 — Terraform Outputs & Attributes Reporting

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

resource "aws_vpc" "development_vpc" {
  cidr_block = "10.0.0.0/16"
  tags = {
    Name: "development"
  }
}

resource "aws_subnet" "dev_subnet_1" {
  vpc_id      = aws_vpc.development_vpc.id
  cidr_block = "10.0.10.0/24"
  availability_zone = "me-central-1a"
  tags = {
    Name: "subnet-1-dev"
  }
}
data "aws_vpc" "existing_vpc" {
  default = true
}
resource "aws_subnet" "dev_subnet_1_existing" {
  vpc_id      = data.aws_vpc.existing_vpc.id
  cidr_block = "172.31.48.0/24"
  availability_zone = "me-central-1a"
  tags = {
    Name: "subnet-1-default"
  }
}
output "dev-vpc-id" {
  value = aws_vpc.development_vpc.id
}
output "dev-subnet-id" {
  value = aws_subnet.dev_subnet_1.id
}
output "dev-vpc-arn" {
  value = aws_vpc.development_vpc.arn
}
output "dev-subnet-arn" {
  value = aws_subnet.dev_subnet_1.arn
}
~
~
~
~
~
~
```

```
Outputs:

dev-subnet-arn = "arn:aws:ec2:me-central-1:443915509636:subnet/subnet-01bfeb495dadc4aa9"
dev-subnet-id = "subnet-01bfeb495dadc4aa9"
dev-vpc-arn = "arn:aws:ec2:me-central-1:443915509636:vpc/vpc-04024e4c880ab1358"
dev-vpc-id = "vpc-04024e4c880ab1358"
```

○ @Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ ☐

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

Outputs:

dev-subnet-arn = "arn:aws:ec2:me-central-1:443915509636:subnet/subnet-01bfeb495dad
dev-subnet-cidr_block = "10.0.10.0/24"
dev-subnet-id = "subnet-01bfeb495dadc4aa9"
dev-subnet-region = "me-central-1a"
dev-subnet-tags_all = tomap({
  "Name" = "subnet-1-dev"
})
dev-subnet-tags_name = "subnet-1-dev"
dev-vpc-arn = "arn:aws:ec2:me-central-1:443915509636:vpc/vpc-04024e4c880ab1358"
dev-vpc-cidr_block = "10.0.0.0/16"
dev-vpc-id = "vpc-04024e4c880ab1358"
dev-vpc-region = "me-central-1"
dev-vpc-tags_all = tomap({
  "Name" = "development"
})
dev-vpc-tags_name = "development"
```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ █

Cleanup — Delete Resources & State Verification

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ terraform destroy
        - dhcp_options_id                           = "dopt-0426dabcc5c93c0f9" -> null
        - enable_dns_hostnames                      = false -> null
        - enable_dns_support                        = true -> null
        - enable_network_address_usage_metrics      = false -> null
        - id                                        = "vpc-04024e4c880ab1358" -> null
        - instance_tenancy                          = "default" -> null
        - ipv6_netmask_length                       = 0 -> null
        - main_route_table_id                       = "rtb-0514cac11257ebda6" -> null
        - owner_id                                  = "443915509636" -> null
        - region                                    = "me-central-1" -> null
        - tags                                      = {
            - "Name" = "development"
          } -> null
        - tags_all                                  = {
            - "Name" = "development"
          } -> null
          # (4 unchanged attributes hidden)
    }

Plan: 0 to add, 0 to change, 3 to destroy.

Changes to Outputs:
  - dev-subnet-arn        = "arn:aws:ec2:me-central-1:443915509636:subnet/subnet-01bfeb495dadc4aa9" -> null
  - dev-subnet-cidr_block = "10.0.10.0/24" -> null
  - dev-subnet-id         = "subnet-01bfeb495dadc4aa9" -> null
  - dev-subnet-region     = "me-central-1a" -> null
  - dev-subnet-tags_all   = {
      - Name = "subnet-1-dev"
    } -> null
  - dev-subnet-tags_name  = "subnet-1-dev" -> null
  - dev-vpc-arn           = "arn:aws:ec2:me-central-1:443915509636:vpc/vpc-04024e4c880ab1358" -> null
  - dev-vpc-cidr_block    = "10.0.0.0/16" -> null
  - dev-vpc-id            = "vpc-04024e4c880ab1358" -> null
  - dev-vpc-region        = "me-central-1" -> null
  - dev-vpc-tags_all      = {
      - Name = "development"
    } -> null
  - dev-vpc-tags_name     = "development" -> null

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

aws_subnet.dev_subnet_1_existing: Destroying... [id=subnet-014bb9108f53cfa35]
aws_subnet.dev_subnet_1: Destroying... [id=subnet-01bfeb495dadc4aa9]
aws_subnet.dev_subnet_1: Destruction complete after 1s
aws_vpc.development_vpc: Destroying... [id=vpc-04024e4c880ab1358]
aws_subnet.dev_subnet_1_existing: Destruction complete after 1s
aws_vpc.development_vpc: Destruction complete after 1s

Destroy complete! Resources: 3 destroyed.
```

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cat terraform.tfstate
cat terraform.tfstate.backup
        },
        {
          "mode": "managed",
          "type": "aws_vpc",
          "name": "development_vpc",
          "provider": "provider[\"registry.terraform.io/hashicorp/aws\"]",
          "instances": [
            {
              "schema_version": 1,
              "attributes": {
                "arn": "arn:aws:ec2:me-central-1:443915509636:vpc/vpc-04024e4c880ab1358"
                "assign_generated_ipv6_cidr_block": false,
                "cidr_block": "10.0.0.0/16",
                "default_network_acl_id": "acl-040674ff1394c131a",
                "default_route_table_id": "rtb-0514cac11257ebda6",
                "default_security_group_id": "sg-05386a7e6f34196fc",
                "dhcp_options_id": "dopt-0426dabcc5c93c0f9",
                "enable_dns_hostnames": false,
                "enable_dns_support": true,
                "enable_network_address_usage_metrics": false,
                "id": "vpc-04024e4c880ab1358",
                "instance_tenancy": "default",
                "ipv4_ipam_pool_id": null,
                "ipv4_netmask_length": null,
                "ipv6_association_id": "",
                "ipv6_cidr_block": "",
                "ipv6_cidr_block_network_border_group": "",
                "ipv6_ipam_pool_id": "",
                "ipv6_netmask_length": 0,
                "main_route_table_id": "rtb-0514cac11257ebda6",
                "owner_id": "443915509636",
                "region": "me-central-1",
                "tags": {
                  "Name": "development"
                },
                "tags_all": {
                  "Name": "development"
                }
              },
              "sensitive_attributes": [],
              "identity_schema_version": 0,
              "identity": {
                "account_id": "443915509636",
                "id": "vpc-04024e4c880ab1358",
                "region": "me-central-1"
              },
              "private": "eyJzY2hlbWFfdmVyc2lvbiI6IjEifQ=="
            }
          ]
        }
      ],
      "check_results": null
}
```