Name: Sadaf Riaz

Roll no: 2023-BSE-077

Class: BSE5-B

Lab 11

Task 0 – Lab Setup (Codespace & GH CLI)

0.1 Repository Creation – taskA_codespace_create_and_list.png

```
PS C:\Users\Lenovo> gh codespace create --repo SadafRiaz-077/CC_SadafRiaz_077_Lab11
  ✓ Codespaces usage for this repository is paid for by SadafRiaz-077
? Choose Machine Type: 2 cores, 8 GB RAM, 32 GB storage
curly-winner-jjp5p9v4vpxvcp479
```

0.2 Codespace Creation & SSH – taskA_codespace_ssh_connected.png

```
PS C:\Users\Lenovo> gh codespace ssh -c curly-winner-jjp5p9v4vpxvcp479
Enter passphrase for key 'C:\Users\Lenovo/.ssh/id_ed25519':
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.

@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

Task 1 – Provider & Basic Variable (Variable Precedence)

1.1 Create main.tf – task1_touch_main_tf.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ touch main.tf
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

1.2 AWS Provider Configuration – task1_main_tf_provider.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ nano main.tf
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

1.3 Terraform Initialization – task1_terraform_init.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (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
commands will detect it and remind you to do so if necessary.
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

1.4 Variable & Output Definition – task1_variable_and_output_added.png

```
 GNU nano 7.2                                                    main.
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
```

1.5 Apply Without Default – task1_apply_prompt_for_var.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve
var.subnet_cidr_block
  Enter a value: 10.0.0.0/24


No changes. Your infrastructure matches the configuration.

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

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

Outputs:

subnet_cidr_block_output = "10.0.0.0/24"
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

1.6 Apply with Default Value – task1_apply_with_default.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

No changes. Your infrastructure matches the configuration.

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

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

Outputs:

subnet_cidr_block_output = "10.0.0.0/24"
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

1.7 Environment Variable (TF_VAR) Set & Apply – task1_env_var_set_and_apply.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ export TF_VAR_subnet_cidr_block=10.0.20.0/24
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

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

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

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

Outputs:

subnet_cidr_block_output = "10.0.20.0/24"
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

1.8 terraform.tfvars Override – task1_terraform_tfvars_and_apply.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ export TF_VAR_subnet_cidr_block=10.0.20.0/24
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

No changes. Your infrastructure matches the configuration.

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

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

Outputs:

subnet_cidr_block_output = "10.0.20.0/24"
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

1.9 -var Override – task1_var_override_with_dash_var.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ nano terraform.tfvars
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

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

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

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

Outputs:

subnet_cidr_block_output = "10.0.30.0/24"
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

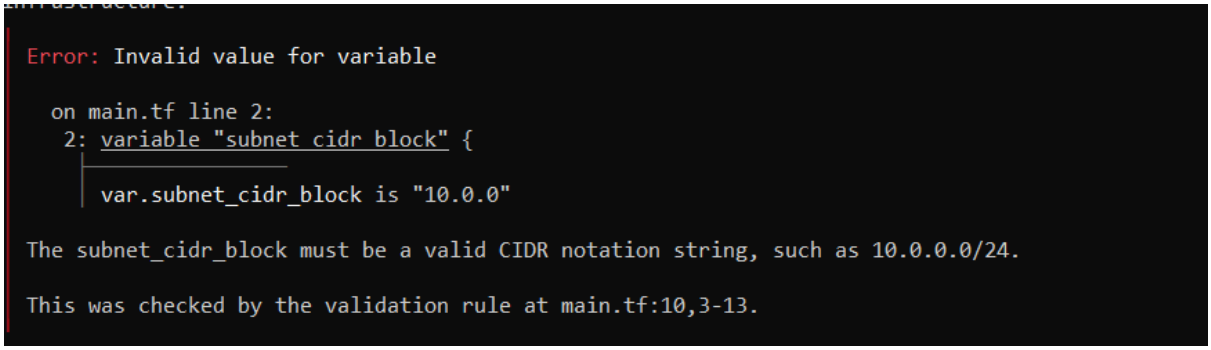
1.10 Unset Environment Variable – task1_printenv_tf_var_and_unset.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve -var "subnet_cidr_block=10.
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"
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

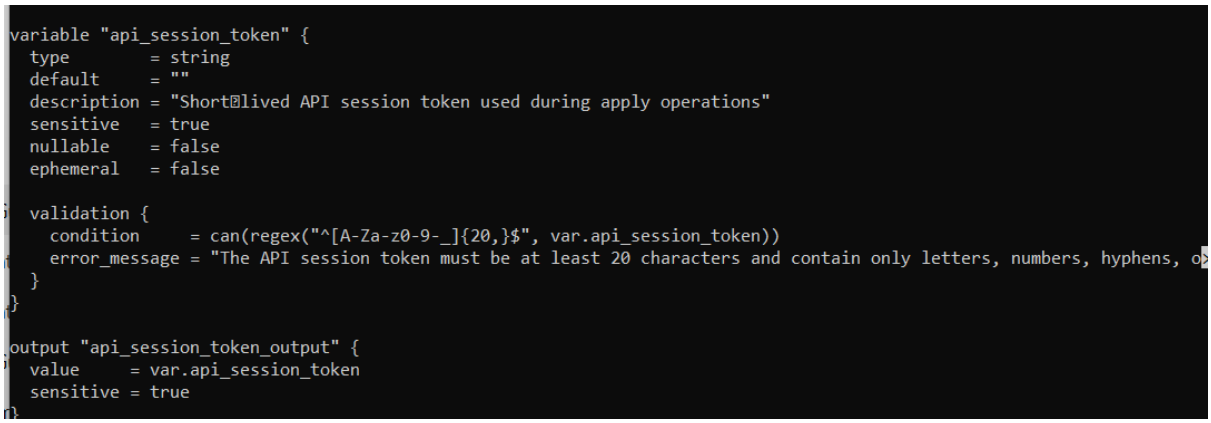
Task 2 – Variable Validation & Sensitive / Ephemeral Variables

2.1 CIDR Variable with Validation – task2_subnet_variable_with_validation.png

2.2 Validation Failure Test – task2_subnet_validation_error.png



2.3 Sensitive Variable (API Token) Added – task2_api_token_variable_added.png



2.4 Sensitive Output Apply – task2_api_token_apply_sensitive.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve -var "
I_session_Token"

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

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

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

Outputs:

api_session_token_output = <sensitive>
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

2.5 Check Terraform State for Sensitive Output – task2_check_terraform_state_api_token.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ cat terraform.tfstate | grep -A 5 api_session_token_output
    "api_session_token_output": {
      "value": "my_API_session_Token",
      "type": "string",
      "sensitive": true
    }
  },
```

2.6 Ephemeral Variable Test – task2_api_token_ephemeral_error.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve -var "subnet_cidr_block=10.0.0"


Changes to Outputs:
  ~ api_session_token_output = (sensitive value)

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

Error: Invalid value for variable

  on main.tf line 8:
   8: 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:16,3-13.

@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

2.7 Default Sensitive Variable Apply – task2_api_token_default_apply.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve -var "api_session_token=my_Secret_API_Token_123"

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"
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

Task 3 – Project-level Variables, Locals & Outputs

3.1 Project Variables Added – task3_variables_added.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ nano main.tf
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

3.2 terraform.tfvars Populated with Subnet ID – task3_terraform_tfvars_populated.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ nano terraform.tfvars
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

3.3 Locals Definition – task3_locals_tf_created.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ nano locals.tf
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

3.4 Outputs Applied – task3_outputs_apply.png

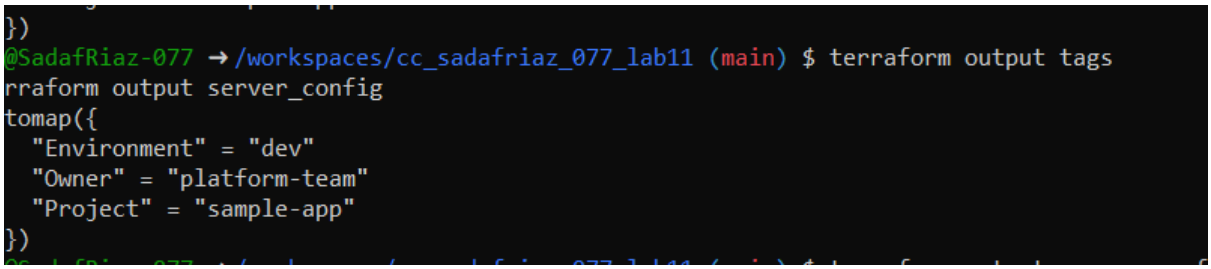
```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  - api_session_token_output = (sensitive value) -> null
  + is_production            = false
  + monitoring_enabled       = true
  + primary_public_subnet    = "404842057573"
  + resource_name            = "lab_work-dev"
  - subnet_cidr_block_output = "10.0.30.0/24" -> null
  + 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:

is_production = false
monitoring_enabled = true
primary_public_subnet = "404842057573"
resource_name = "lab_work-dev"
subnet_count = 3
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

Task 4 – Maps & Objects

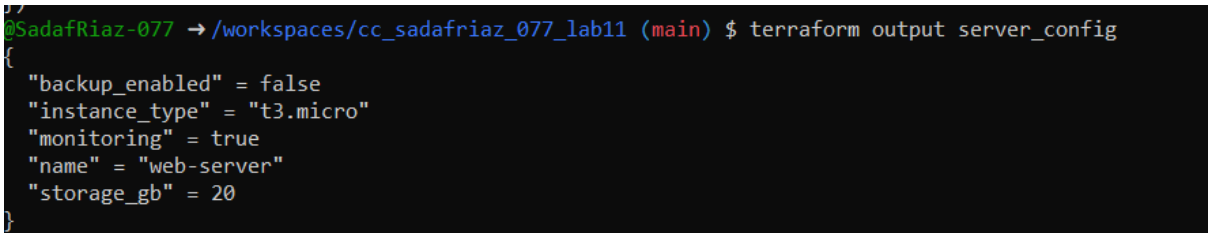
4.1 Tags Map Variable Added – task4_tags_variable_added.png

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

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

4.2 Tags Output Applied – task4_tags_output.png

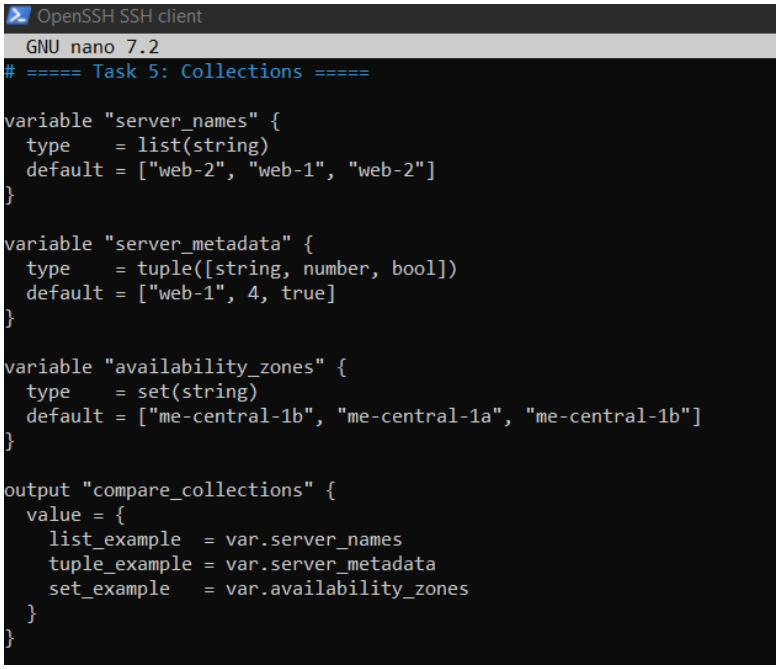


4.3 Server Config Object Variable – task4_server_config_output.png



Task 5 – Collections: List, Tuple, Set & Mutations via Locals

5.1 Define Collection Variables – task5_collections_defined.png



5.2 Compare Collections Output – task5_compare_collections.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ nano main.tf
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  + compare_collections = {
      + list_example  = [
          + "web-2",
          + "web-1",
          + "web-2",
        ]
      + set_example   = [
          + "me-central-1a",
          + "me-central-1b",
        ]
      + tuple_example = [
          + "web-1",
          + 4,
          + true,
        ]
    }
  - server_config       = {
      - backup_enabled = false
      - instance_type  = "t3.micro"
      - monitoring     = true
      - name           = "web-server"
      - storage_gb     = 20
    } -> null
  - tags                = {
      - Environment = "dev"
      - Owner       = "platform-team"
      - Project     = "sample-app"
    } -> null

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:

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,
  ]
}
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

5.3 Locals Mutation (setunion) – task5_locals_mutations.png

```
OpenSSH SSH client
  GNU nano 7.2
locals {
  mutated_list  = setunion(var.server_names, ["web-3"])
  mutated_tuple = setunion(var.server_metadata, ["web-2"])
  mutated_set   = setunion(var.availability_zones, ["me-central-1c"])
}
```

5.4 Mutation Comparison Output – task5_mutation_comparison.png

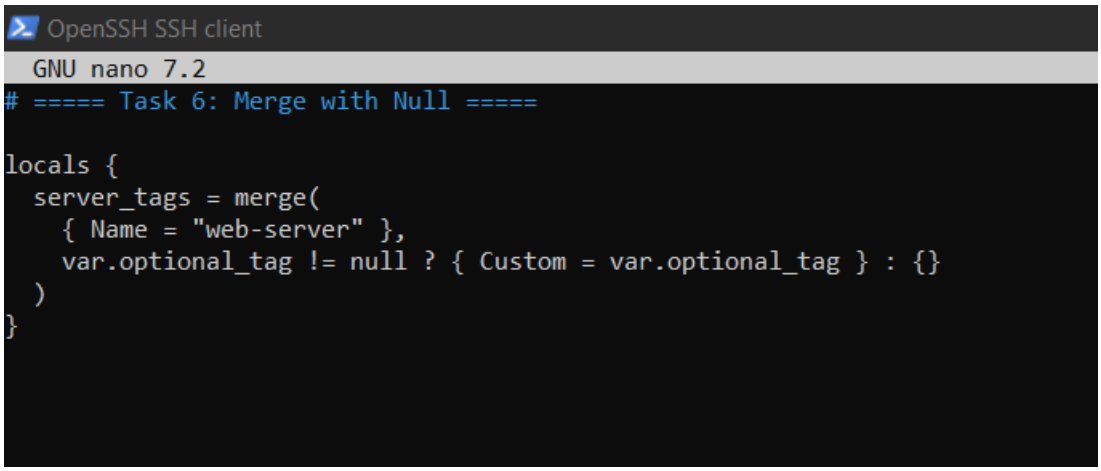
```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  + mutation_comparison = {
      + mutated_tuple  = [
          + "4",
          + "true",
          + "web-1",
          + "web-2",
        ]
      + original_tuple = [
          + "web-1",
          + 4,
          + true,
        ]
    }

You can apply this plan to save these new output values to the Terraform state, without changing an

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

Outputs:

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,
  ]
}
mutation_comparison = {
  "mutated_tuple" = toset([
    "4",
    "true",
```

Task 6 – Null, Any Type & Dynamic Values

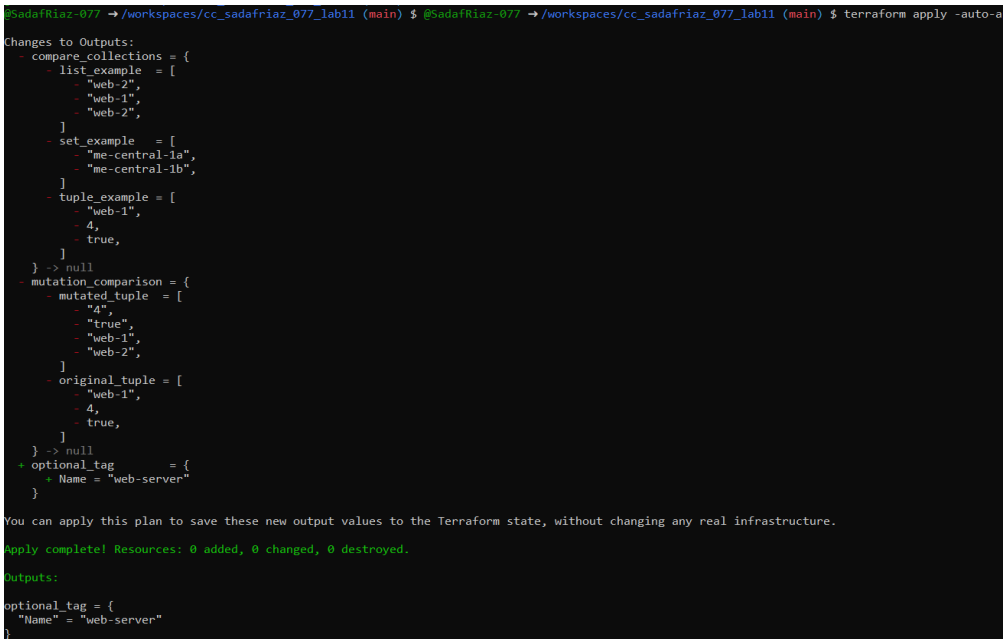
6.1 Optional Tag Variable – task6_optional_tag_variable.png

```
OpenSSH SSH client
  GNU nano 7.2
# ===== Task 6: Null Variable =====

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

output "optional_tag" {
  value = local.server_tags
}

# ===== Any Type Variable =====

variable "dynamic_value" {
  type        = any
  description = "A variable that can accept any data type"
  default     = null
}

output "value_received" {
  value = var.dynamic_value
}
```

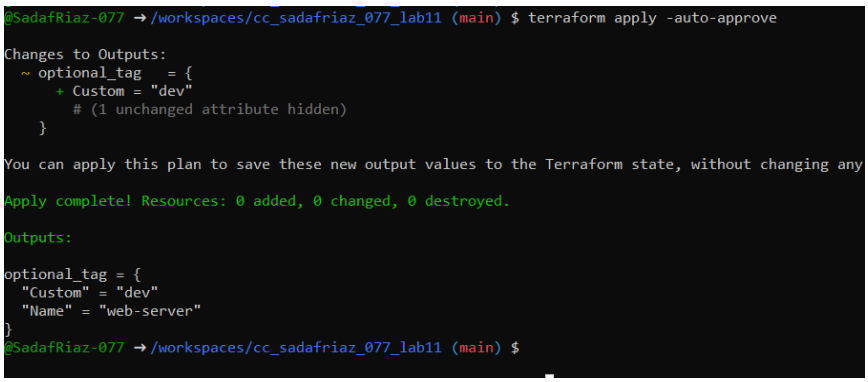
6.2 Merge Tags in Locals – task6_locals_merge.png



6.3 Optional Tag Output (No Value) – task6_optional_tag_no_value.png



6.4 Optional Tag Output (With Value) – task6_optional_tag_with_value.png



6.5 Dynamic Value – String – task6_dynamic_value_string.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  + value_received = "hello"

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:

optional_tag = {
  "Custom" = "dev"
  "Name" = "web-server"
}
value_received = "hello"
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

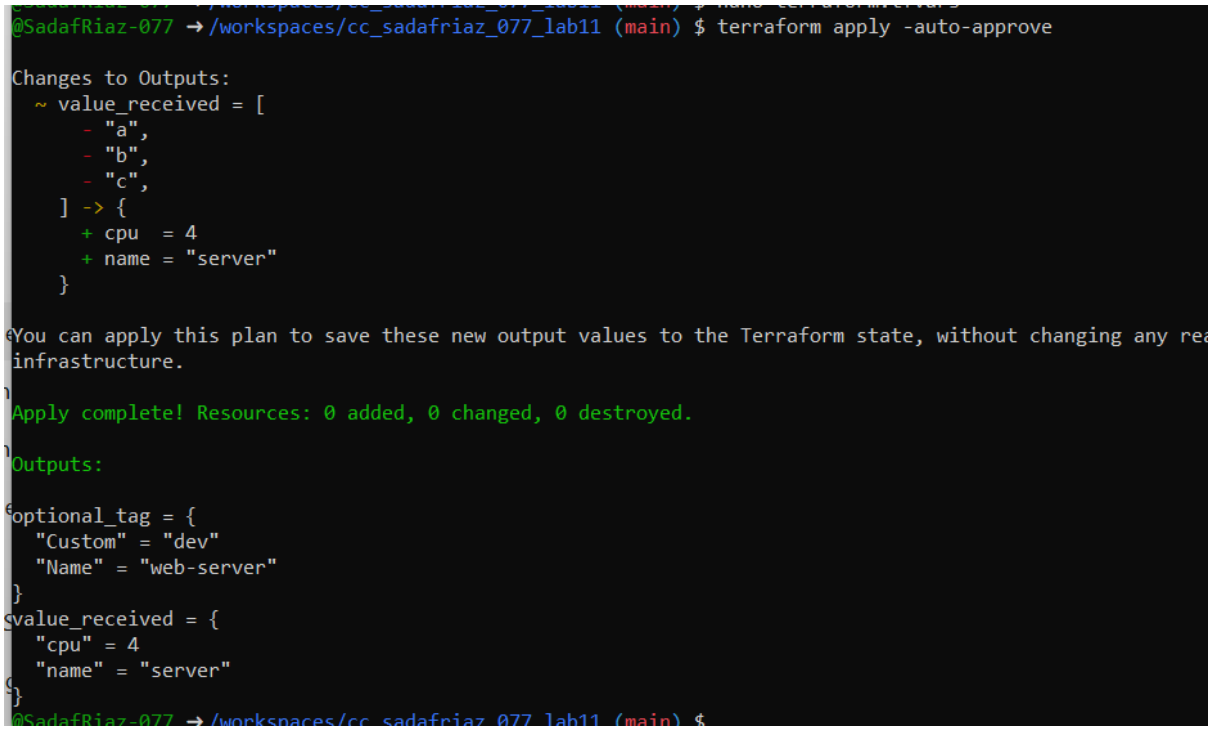
6.6 Dynamic Value – Number – task6_dynamic_value_number.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  ~ value_received = "hello" -> 42

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

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

Outputs:

optional_tag = {
  "Custom" = "dev"
  "Name" = "web-server"
}
value_received = 42
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

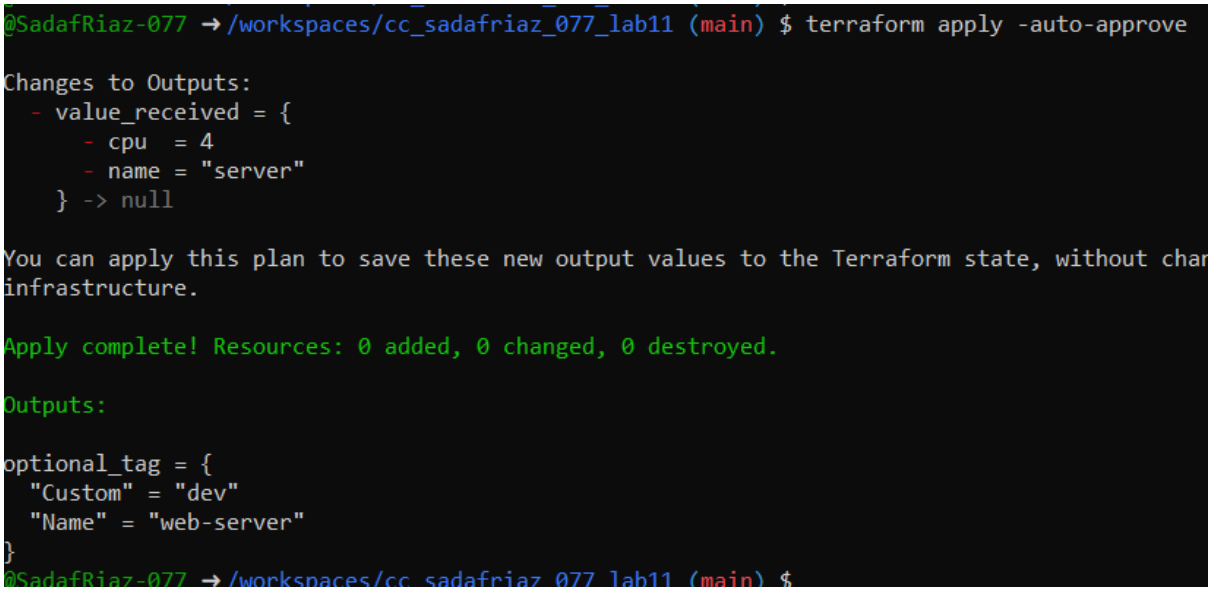
6.7 Dynamic Value – List – task6_dynamic_value_list.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve

Changes to Outputs:
  ~ value_received = 42 -> [
      + "a",
      + "b",
      + "c",
    ]

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

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

Outputs:

optional_tag = {
  "Custom" = "dev"
  "Name" = "web-server"
}
value_received = [
  "a",
  "b",
  "c",
]
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

6.8 Dynamic Value – Map/Object – task6_dynamic_value_map.png

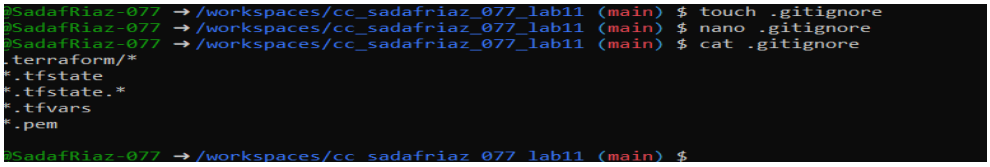


6.9 Dynamic Value – Null – task6_dynamic_value_null.png



Task 7 – Git Ignore

7.1 Create .gitignore – task7_gitignore_created.png



Task 8 – AWS Infrastructure (VPC, Subnet, IGW, Routing)

8.1 Cleanup & Reset Files – task8_clean_files.png

```
  GNU nano 7.2
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}
```

8.2 Variables Recreated in main.tf – task8_variables_recreated.png

```
  GNU nano 7.2                                                    ma
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" {}
```

8.3 VPC Resource Added – task8_vpc_resources_added.png

```
  GNU nano 7.2
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"
  }
}
```

8.4 Subnet Resource Added – task8_subnet_resources_added.png

```
}
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"
  }
}
```

8.5 terraform.tfvars Populated with VPC Values – task8_terraform_tfvars_vpc_values.png

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

8.6 Apply VPC & Subnet – task8_vpc_subnet_apply.png

```
# aws_vpc.myapp_vpc will be created
+ resource "aws_vpc" "myapp_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" = "dev-vpc"
      }
    + tags_all                             = {
        + "Name" = "dev-vpc"
      }
  }

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

hanges to Outputs:
 - optional_tag = {
     - Custom = "dev"
     - Name   = "web-server"
   } -> null
ws_vpc.myapp_vpc: Creating...
ws_vpc.myapp_vpc: Creation complete after 3s [id=vpc-00963b4a5bcf35790]
ws_subnet.myapp_subnet_1: Creating...
ws_subnet.myapp_subnet_1: Creation complete after 1s [id=subnet-0099fd10603a132a0]

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

8.7 IGW & Custom Route Table Before Apply – task8_igw_route_table_before_apply.png

```
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"
  }
}
```

8.8 IGW & Route Table After Apply – task8_igw_route_table_after_apply.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-00963b4a5bcf35790]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-0099fd10603a132a0]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with
  + 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-00963b4a5bcf35790"
    }

  # 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-00963b4a5bcf35790"
    }

Plan: 2 to add, 0 to change, 0 to destroy.
aws_internet_gateway.myapp_igw: Creating...
aws_internet_gateway.myapp_igw: Creation complete after 2s [id=igw-031bedab9c250606d]
aws_route_table.myapp_route_table: Creating...
aws_route_table.myapp_route_table: Creation complete after 1s [id=rtb-0e6328e001a998f5e]
```

8.9 Route Table Association Applied – task8_association_apply.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-00963b4a5bcf35790]
aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-031bedab9c250606d]50606d]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-0099fd10603a132a0]
aws_route_table.myapp_route_table: Refreshing state... [id=rtb-0e6328e001a998f5e]998f5e]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
  + create-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve
^[[A^[[A^[[A^[[A^[[A^[[A^[[Aaws_vpc.myapp_vpc: Refreshing state... [id=vpc-00963b4a5bcf35790]
Terraform will perform the following actions:ate... [id=igw-031bedab9c250606d]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-0099fd10603a132a0]
  # aws_route_table_association.a_rtb_subnet will be createdtb-0e6328e001a998f5e]
  + resource "aws_route_table_association" "a_rtb_subnet" {
      + id             = (known after apply)configuration.
      + region         = "me-central-1"
      + route_table_id = "rtb-0e6328e001a998f5e"against your configuration and found no differences, so no changes are n
      + subnet_id      = "subnet-0099fd10603a132a0"
    } complete! Resources: 0 added, 0 changed, 0 destroyed.
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ aws --version
Plan: 1 to add, 0 to change, 0 to destroy.
aws_route_table_association.a_rtb_subnet: Creating...
aws_route_table_association.a_rtb_subnet: Creation complete after 0s [id=rtbassoc-0435c193ae6bc9b77]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

8.10 Default Route Table Resource Added – task8_default_route_table.png

```
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"
  }
}
```

8.11 Default Route Table Applied – task8_default_route_table_apply.png

```
  # 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-1:404842057573:route-table/rtb-0e6328e001a998f5e" -> null
      - id               = "rtb-0e6328e001a998f5e" -> null
      - owner_id         = "404842057573" -> null
      - propagating_vgws = [] -> null
      - region           = "me-central-1" -> null
      - route            = [
          - {
              - cidr_block                = "0.0.0.0/0"
              - gateway_id                = "igw-031bedab9c250606d"
                # (11 unchanged attributes hidden)
            },
        ] -> null
      - tags             = {
          - "Name" = "dev-rt"
        } -> null
      - tags_all         = {
          - "Name" = "dev-rt"
        } -> null
      - vpc_id           = "vpc-00963b4a5bcf35790" -> 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-0435c193ae6bc9b77" -> null
      - region         = "me-central-1" -> null
      - route_table_id = "rtb-0e6328e001a998f5e" -> null
      - subnet_id      = "subnet-0099fd10603a132a0" -> 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-0435c193ae6bc9b77]
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-0e6328e001a998f5e]
aws_route_table.myapp_route_table: Destruction complete after 1s
aws_default_route_table.main_rt: Creation complete after 1s [id=rtb-078ddfda594446d71]
```

Task 9 – Security Group, Key Pair, EC2 Instance, User_data & Nginx

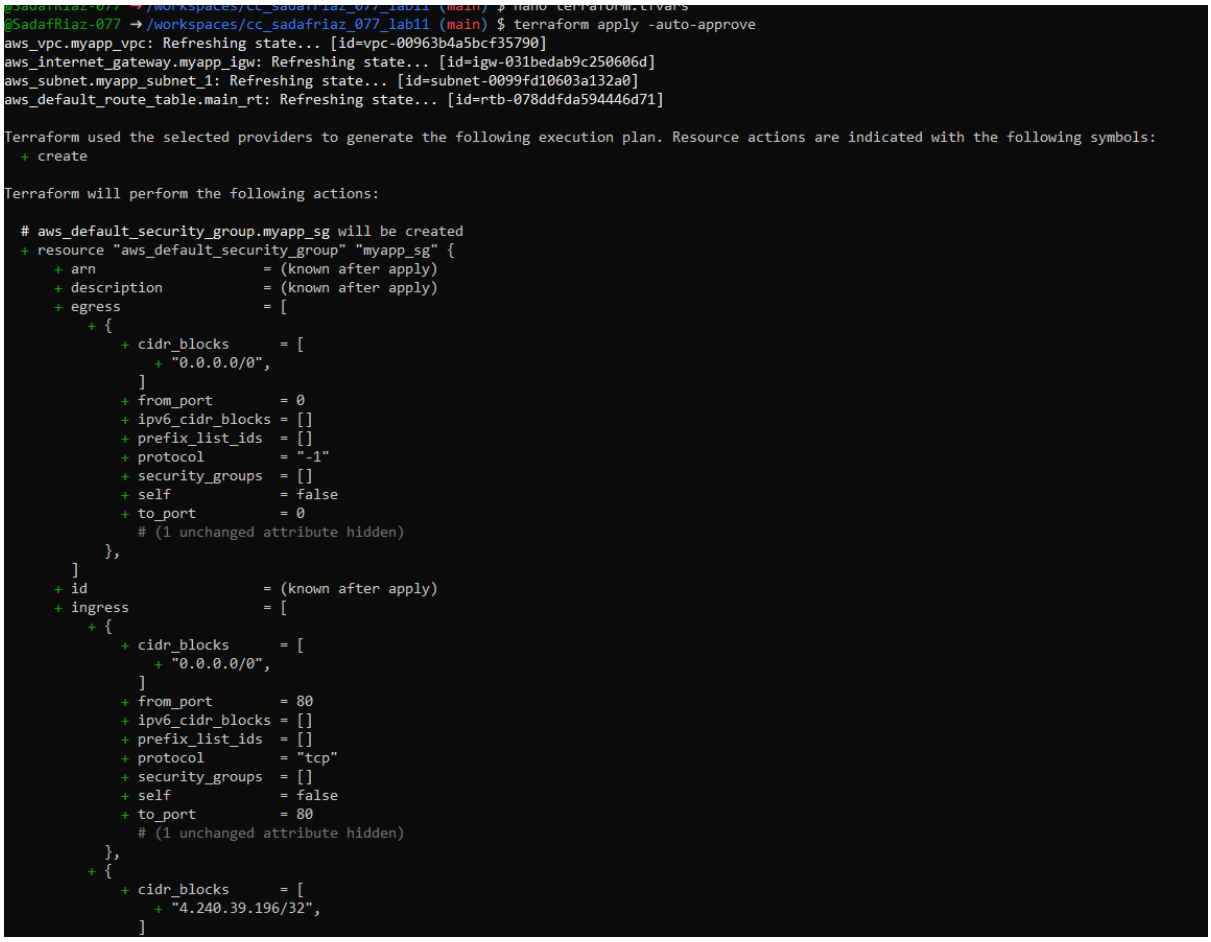
9.1 my_ip Variable Added – task9_my_ip_variable_added.png

```
resource "aws_vpc" "myapp_vpc" {
  cidr_block = var.vpc_cidr_block
  tags = {
     Name = "${var.env_prefix}-vpc"
  }
}
resource "aws_subnet" "myapp_subnet_1" {
  vpc_id            = aws_vpc.myapp_vpc.id
  cidr_block        = var.subnet_cidr_block
  availability_zone = var.availability_zone
  tags = {
     Name = "${var.env_prefix}-subnet-1"
  }
}
resource "aws_internet_gateway" "myapp_igw" {
  vpc_id = aws_vpc.myapp_vpc.id
  tags = {
     Name = "${var.env_prefix}-igw"
  }
}


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

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

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

variable "my_ip" {}
variable "instance_type" {}
```

9.2 Public IP Retrieved & terraform.tfvars Updated – task9_public_ip_curl.png

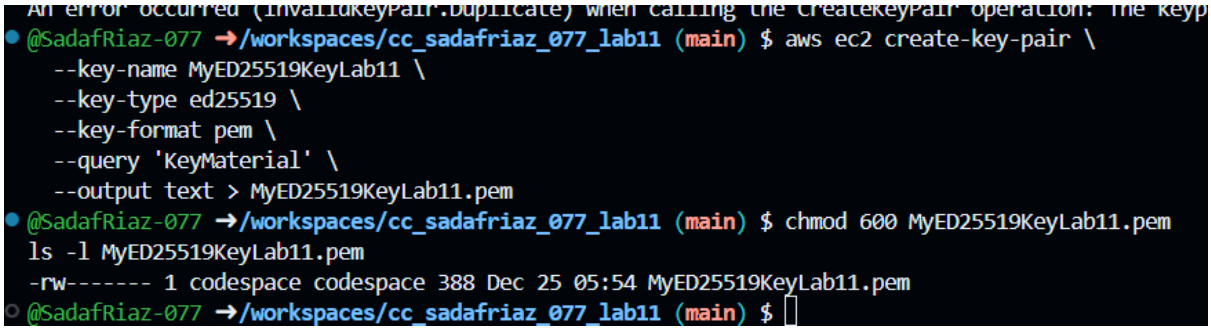
```
variable "my_ip" {}
variable "instance_type" {}
variable "availability_zone" {}
variable "env_prefix" {}
resource "aws_default_security_group" "myapp_sg" {
  vpc_id = aws_vpc.myapp_vpc.id

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

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

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

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

9.3 Security Group Created & Applied – task9_security_group_apply.png



```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-00963b4a5bcf35790]
aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-031bedab9c250606d]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-0099fd10603a132a0]
aws_default_route_table.main_rt: Refreshing state... [id=rtb-078ddfda594446d71]

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_default_security_group.myapp_sg will be created
  + resource "aws_default_security_group" "myapp_sg" {
      + arn                    = (known after apply)
      + description            = (known after apply)
      + egress                 = [
          + {
              + cidr_blocks      = [
                  + "0.0.0.0/0",
                ]
              + from_port        = 0
              + ipv6_cidr_blocks = []
              + prefix_list_ids  = []
              + protocol         = "-1"
              + security_groups  = []
              + self             = false
              + to_port          = 0
                # (1 unchanged attribute hidden)
            },
        ]
      + id                     = (known after apply)
      + ingress                = [
          + {
              + cidr_blocks      = [
                  + "0.0.0.0/0",
                ]
              + from_port        = 80
              + ipv6_cidr_blocks = []
              + prefix_list_ids  = []
              + protocol         = "tcp"
              + security_groups  = []
              + self             = false
              + to_port          = 80
                # (1 unchanged attribute hidden)
            },
          + {
              + cidr_blocks      = [
                  + "4.240.39.196/32",
                ]
```

9.4 Key Pair Created & Saved Locally – task9_keypair_created_and_saved.png

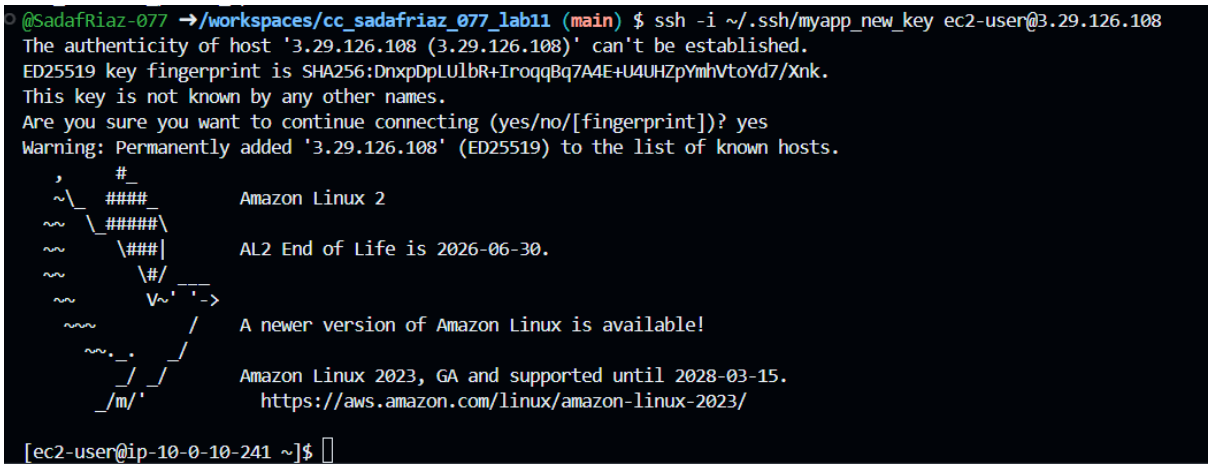


```
An error occurred (InvalidKeyPair.Duplicate) when calling the CreateKeyPair operation: The keyp
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ aws ec2 create-key-pair \
  --key-name MyED25519KeyLab11 \
  --key-type ed25519 \
  --key-format pem \
  --query 'KeyMaterial' \
  --output text > MyED25519KeyLab11.pem
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ chmod 600 MyED25519KeyLab11.pem
ls -l MyED25519KeyLab11.pem
-rw------- 1 codespace codespace 388 Dec 25 05:54 MyED25519KeyLab11.pem
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $
```

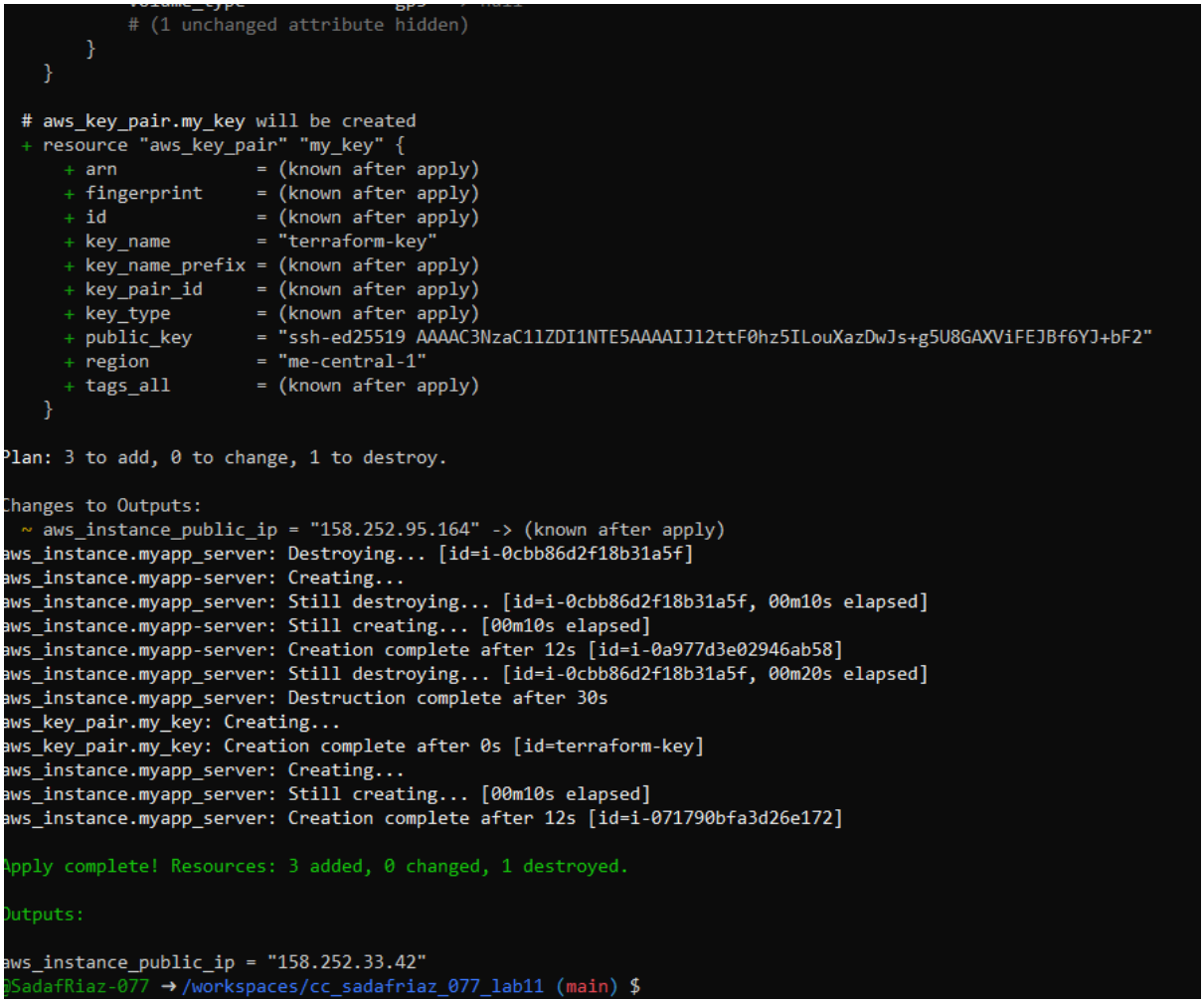
9.5 EC2 Instance Resource Added – task9_instance_type_set.png

```
@SadafRiaz-077 ➜ /workspaces/cc_sadafriaz_077_lab11 (main) $ terraform apply -auto-approve
aws_key_pair.ssh_key: Creating...
aws_key_pair.ssh_key: Creation complete after 0s [id=myapp-new-key]
aws_instance.myapp-server: Creating...
aws_instance.myapp-server: Still creating... [00m10s elapsed]
aws_instance.myapp-server: Creation complete after 12s [id=i-02321555cae21c838]

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

Outputs:
```

9.7 SSH into EC2 – task9_ssh_into_ec2.png



9.8 Terraform-managed SSH Key Pair & Update – task9_ssh_keypair_and_ssh.png



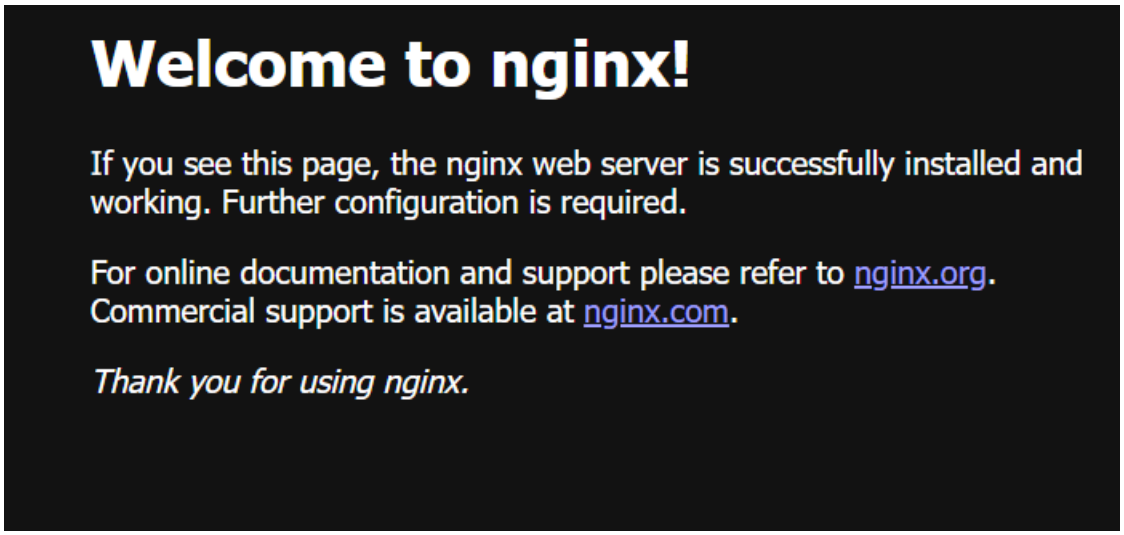
9.9 Nginx Installed via Inline User_data – task9_nginx_local_curl.png

```
@SadafRiaz-077 ➜/workspaces/cc_sadafriaz_077_lab11 (main) $ ssh -i ~/.ssh/myapp_new_key ec2-user@3.29.126.108
Dec 25 09:59:44 ip-10-0-10-241.me-central-1.compute.internal nginx[2536]: nginx: the configuration file /etc/ngin...ok
Dec 25 09:59:44 ip-10-0-10-241.me-central-1.compute.internal nginx[2536]: nginx: configuration file /etc/nginx/ng...ul
Dec 25 09:59:44 ip-10-0-10-241.me-central-1.compute.internal systemd[1]: Started The nginx HTTP and reverse proxy...r.
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-10-0-10-241 ~]$ 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-10-0-10-241 ~]$
```


```
@SadafRiaz-077 ➜/workspaces/cc_sadafriaz_077_lab11 (main) $ ssh -i ~/.ssh/myapp_new_key ec2-user@3.29.126.108
Dec 25 09:59:44 ip-10-0-10-241.me-central-1.compute.internal nginx[2536]: nginx: the configuration file /etc/ngin...ok
Dec 25 09:59:44 ip-10-0-10-241.me-central-1.compute.internal nginx[2536]: nginx: configuration file /etc/nginx/ng...ul
Dec 25 09:59:44 ip-10-0-10-241.me-central-1.compute.internal systemd[1]: Started The nginx HTTP and reverse proxy...r.
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-10-0-10-241 ~]$ 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-10-0-10-241 ~]$
```

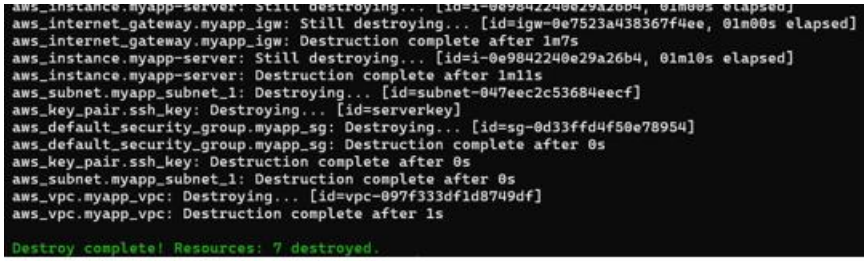

9.10 Nginx Verified in Browser – task9_nginx_browser_page.png

9.10 Nginx Verified in Browser – task9_nginx_browser_page.png

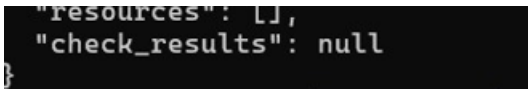


Cleanup

Cleanup – Terraform Destroy – cleanup_destroy.png



Cleanup – Verify State Files – cleanup_state_files.png



Cleanup – Verify No Secrets – cleanup_verify_no_secrets.png