LAB EXAM

NAME: SHUMAIL ZAHRA

REGISTRATION #: 2023-BSE-061

DEPARTMENT: BSE(5B)

LAB TASK

q1_create_group

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam create-group --group-name SoftwareEngineering
{
    "Group": {
        "Path": "/",
        "GroupName": "SoftwareEngineering",
        "GroupId": "AGPARC5V6TLZXQKHZDWBX",
        "Arn": "arn:aws:iam::075006647027:group/SoftwareEngineering",
        "CreateDate": "2026-01-19T07:45:39+00:00"
    }
}
```

q1_group_details

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam get-group --group-name SoftwareEngineering
{
    "Users": [],
    "Group": {
        "Path": "/",
        "GroupName": "SoftwareEngineering",
        "GroupId": "AGPARC5V6TLZXQKHZDWBX",
        "Arn": "arn:aws:iam::075006647027:group/SoftwareEngineering",
        "CreateDate": "2026-01-19T07:45:39+00:00"
    }
}
```

q1_create_user

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam create-user --user-name Shumail
{
    "User": {
        "Path": "/",
        "UserName": "Shumail",
        "UserId": "AIDARC5V6TLZYTZZZYNN4",
        "Arn": "arn:aws:iam::075006647027:user/Shumail",
        "CreateDate": "2026-01-19T07:45:55+00:00"
    }
}
```

q1_user_details

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam get-user --user-name Shumail
{
    "User": {
        "Path": "/",
        "UserName": "Shumail",
        "UserId": "AIDARC5V6TLZYTZZZYNN4",
        "Arn": "arn:aws:iam::075006647027:user/Shumail",
        "CreateDate": "2026-01-19T07:45:55+00:00"
    }
}
```

q1_add_user_to_group

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam add-user-to-group \
  --group-name SoftwareEngineering \
  --user-name Shumail
@23-22411-061-rgb →/workspaces/Lab_exam (main) $
```

0

q1_group_membership

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam get-group --group-name SoftwareEngineering
{
    "Users": [
        {
            "Path": "/",
            "UserName": "Shumail",
            "UserId": "AIDARC5V6TLZYTZZZYNN4",
            "Arn": "arn:aws:iam::075006647027:user/Shumail",
            "CreateDate": "2026-01-19T07:45:55+00:00"
        }
    ],
    "Group": {
        "Path": "/",
        "GroupName": "SoftwareEngineering",
        "GroupId": "AGPARC5V6TLZXQKHZDWBX",
        "Arn": "arn:aws:iam::075006647027:group/SoftwareEngineering",
        "CreateDate": "2026-01-19T07:45:39+00:00"
    }
}
```

q1_find_admin_policy

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam list-policies --scope AWS --query "Policies[?PolicyName=='Ad
ministratorAccess']"
[
    {
        "PolicyName": "AdministratorAccess",
        "PolicyId": "ANPAIWMBCKSKIEE64ZLYK",
        "Arn": "arn:aws:iam::aws:policy/AdministratorAccess",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 1,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2015-02-06T18:39:46+00:00",
        "UpdateDate": "2015-02-06T18:39:46+00:00"
    }
]
@23-22411-061-rgb →/workspaces/Lab_exam (main) $
```

q1_attach_admin_policy

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam list-policies --scope AWS --query "Policies[?PolicyName=='AdministratorAccess']"
[
    {
        "PolicyName": "AdministratorAccess",
        "PolicyId": "ANPAIWMBCKSKIEE64ZLYK",
        "Arn": "arn:aws:iam::aws:policy/AdministratorAccess",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 1,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2015-02-06T18:39:46+00:00",
        "UpdateDate": "2015-02-06T18:39:46+00:00"
    }
]
```

q1_attach_admin_policy

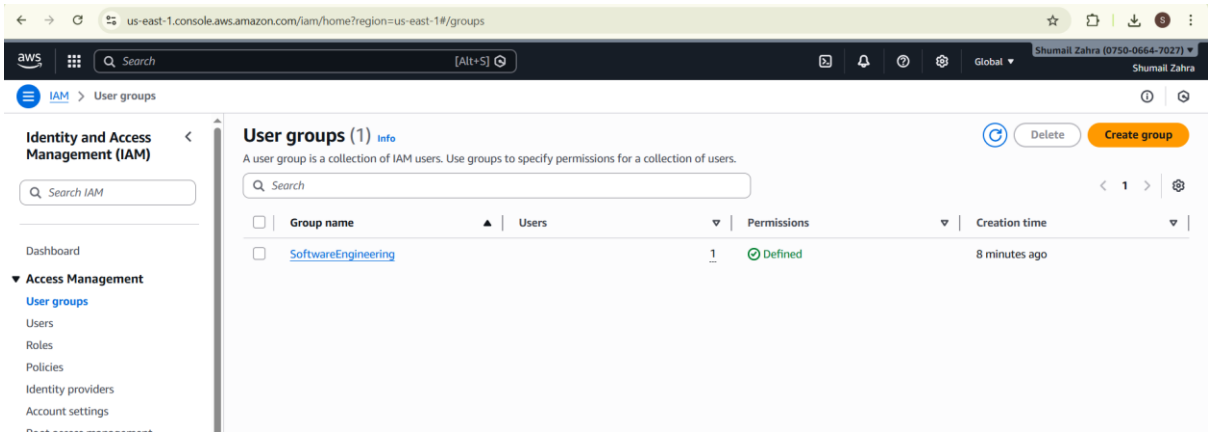
```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam attach-group-policy \
  --group-name SoftwareEngineering \
  --policy-arn arn:aws:iam::aws:policy/AdministratorAccess
```

q1_list_group_policies

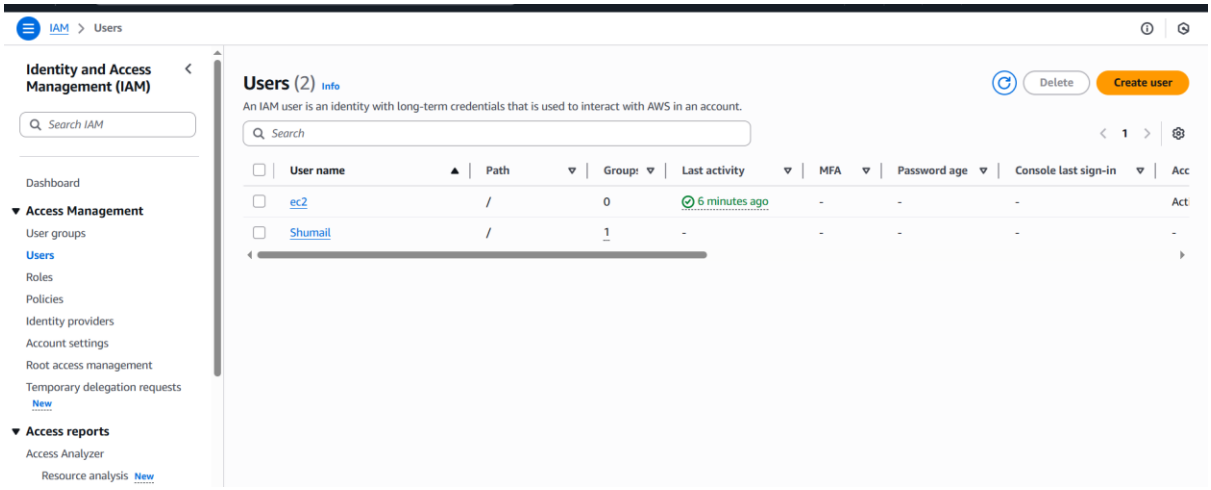
```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam list-attached-group-policies --group-name SoftwareEngig
{
    "AttachedPolicies": [
        {
            "PolicyName": "AdministratorAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
        }
    ]
}
@23-22411-061-rgb →/workspaces/Lab_exam (main) $
```

0 0

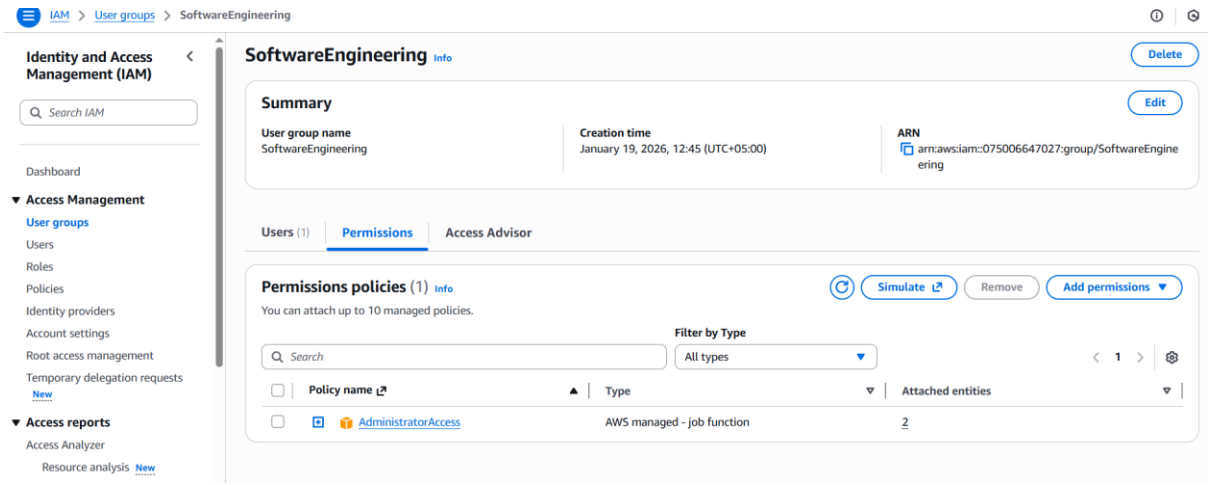
q1_console_group



q1_console_user_in_group



q1_console_group_policy



Q2

q2_provider

```
provider "aws" {
  region = "us-east-1"
}
~
~
~
~
~
~
~
```

q2_variables

```
variable "vpc_cidr_block" {
  type = string
}

variable "subnet_cidr_block" {
  type = string
}

variable "availability_zone" {
  type = string
}

variable "env_prefix" {
  type = string
}

variable "instance_type" {
  type = string
}
```

q2_vpc_subnet

```
provider "aws" {
  region = "us-east-1"
}
resource "aws_vpc" "myapp_vpc" {
  cidr_block = var.vpc_cidr_block

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

resource "aws_subnet" "myapp_subnet" {
  vpc_id                  = aws_vpc.myapp_vpc.id
  cidr_block              = var.subnet_cidr_block
  availability_zone       = var.availability_zone
  map_public_ip_on_launch = true

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

q2_igw_route_table

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

resource "aws_default_route_table" "myapp_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"
  }
}
```

q2_http_and_locals

```
  tags = {
    Name = "${var.env_prefix}-rt"
  }
}
data "http" "my_ip" {
  url = "https://icanhazip.com"
}

locals {
  my_ip = "${chomp(data.http.my_ip.response_body)}/32"
}
```

q2_default_sg

```
  my_ip = "${chomp(data.http.my_ip.response_body)}/32"
}
resource "aws_default_security_group" "default_sg" {
  vpc_id = aws_vpc.myapp_vpc.id

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

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

  ingress {
    from_port   = 443
    to_port     = 443
    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}-default-sg"
```

```
    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}-default-sg"
  }
}
```

q2_keypair

```
}
resource "aws_key_pair" "serverkey" {
  key_name   = "serverkey"
  public_key = file("${path.module}/serverkey.pub")
}
```

q2_ec2_resource

```
resource "aws_instance" "myapp_ec2" {
  ami = "ami-0c2b8ca1dad447f8a"
  instance_type               = var.instance_type
  subnet_id                   = aws_subnet.myapp_subnet.id
  vpc_security_group_ids      = [aws_default_security_group.default_sg.
  availability_zone           = var.availability_zone
  associate_public_ip_address = true
  key_name                    = aws_key_pair.serverkey.key_name
:wq!
```

q2_entry_script

```
#!/bin/bash
dnf update -y
dnf install -y nginx openssl

mkdir -p /etc/nginx/ssl

openssl req -x509 -nodes -days 365 \
  -newkey rsa:2048 \
  -keyout /etc/nginx/ssl/nginx.key \
  -out /etc/nginx/ssl/nginx.crt \
  -subj "/CN=Terraform-Nginx"

cat <<EOF > /etc/nginx/conf.d/default.conf
server {
    listen 80;
    return 301 https://\$host\$request_uri;
}

server {
    listen 443 ssl;
    ssl_certificate /etc/nginx/ssl/nginx.crt;
    ssl_certificate_key /etc/nginx/ssl/nginx.key;

    location / {
        root /usr/share/nginx/html;
        index index.html;
    }
}
EOF

echo "<h1>This is Shumail's Terraform environment</h1>" > /usr/share/nginx/html/index.html

systemctl enable nginx
systemctl restart nginx
```

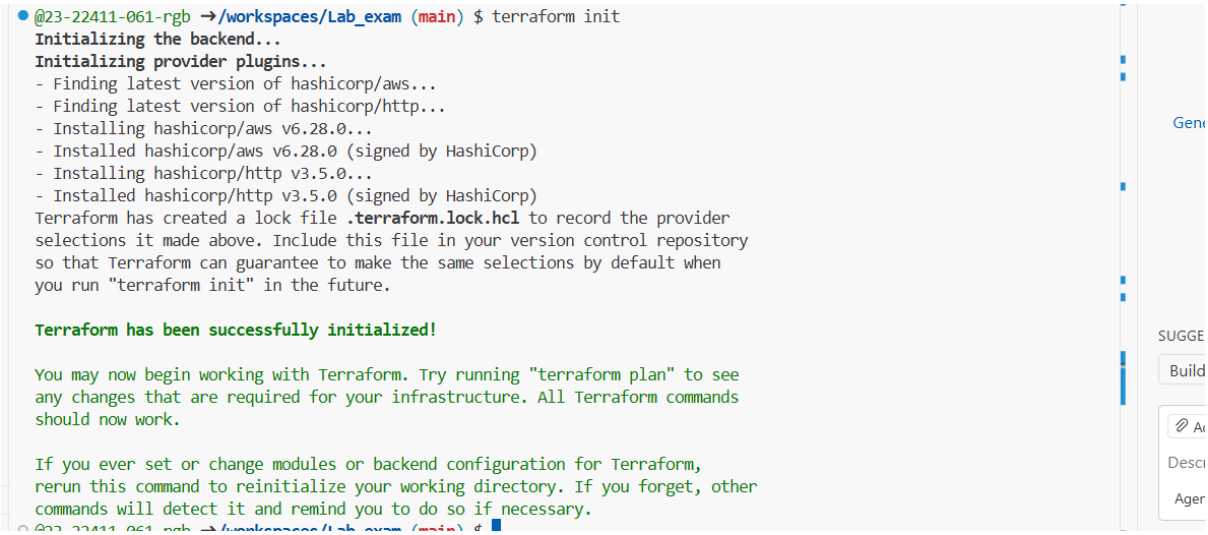
q2_output_block

```
output "ec2_public_ip" {
  value = aws_instance.myapp_ec2.public_ip
}
~
~
~
~
~
~
```

q2_tfvars_or_vars

```
vpc_cidr_block    = "10.0.0.0/16"
subnet_cidr_block = "10.0.10.0/24"
availability_zone = "us-east-1a"
env_prefix        = "dev"
instance_type     = "t3.micro"

~
~
~
~
~
~
~
~
~
~
```

q2_terraform_init



q2_terraform_plan

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $ terraform plan
      + 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                               = "us-east-1"
      + tags                                 = {
          + "Name" = "dev-vpc"
        }
      + tags_all                             = {
          + "Name" = "dev-vpc"
        }
    }

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

Changes to Outputs:
  + ec2_public_ip = (known after apply)

─────────────────────────────────────────────────────────────────────────────

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if
```

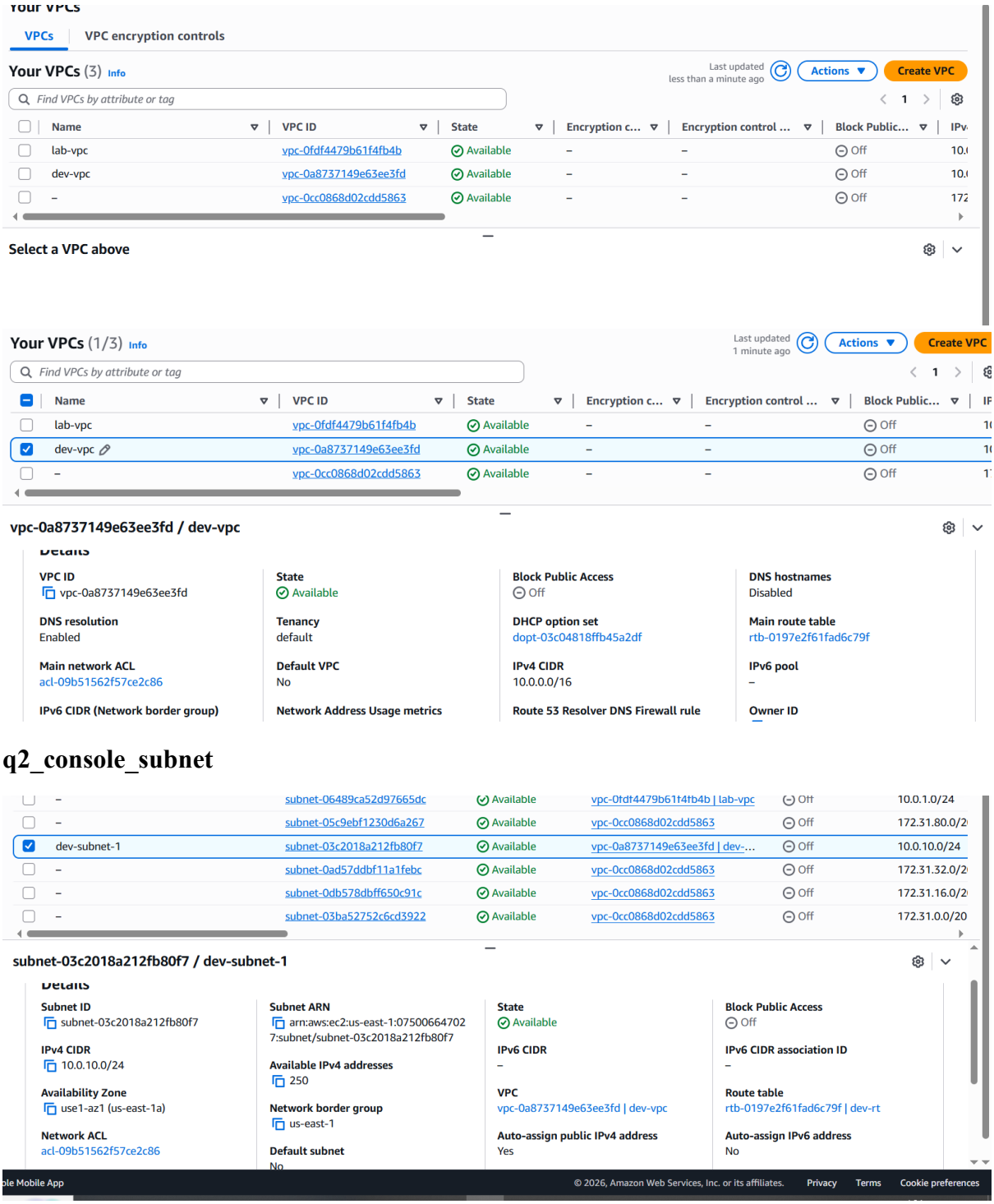
q2_terraform_apply

```
@23-22411-061-rgb ➜/workspaces/Lab_exam (main) $ terraform apply

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + primary_network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

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

Changes to Outputs:
  + ec2_public_ip = (known after apply)

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_instance.myapp_ec2: Creating...
aws_instance.myapp_ec2: Still creating... [00m10s elapsed]
aws_instance.myapp_ec2: Creation complete after 16s [id=i-0452b0ee8f31b4e2d]

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

Outputs:

ec2_public_ip = "3.238.32.182"
@23-22411-061-rgb ➜/workspaces/Lab_exam (main) $
```

q2_terraform_output

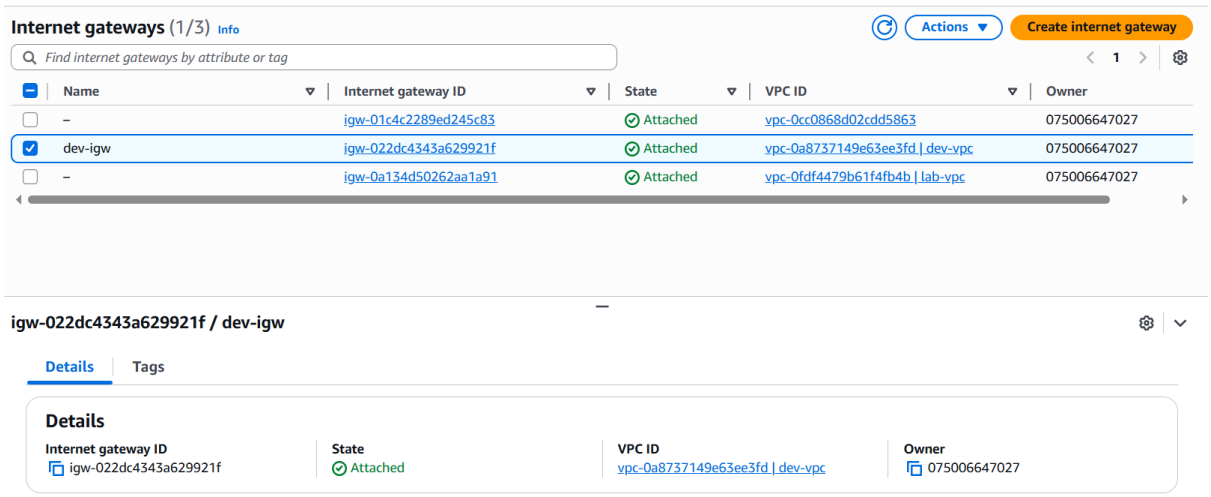
```
@23-22411-061-rgb ➜/workspaces/Lab_exam (main) $ terraform output
ec2_public_ip = "3.238.32.182"
```

q2_console_vpc

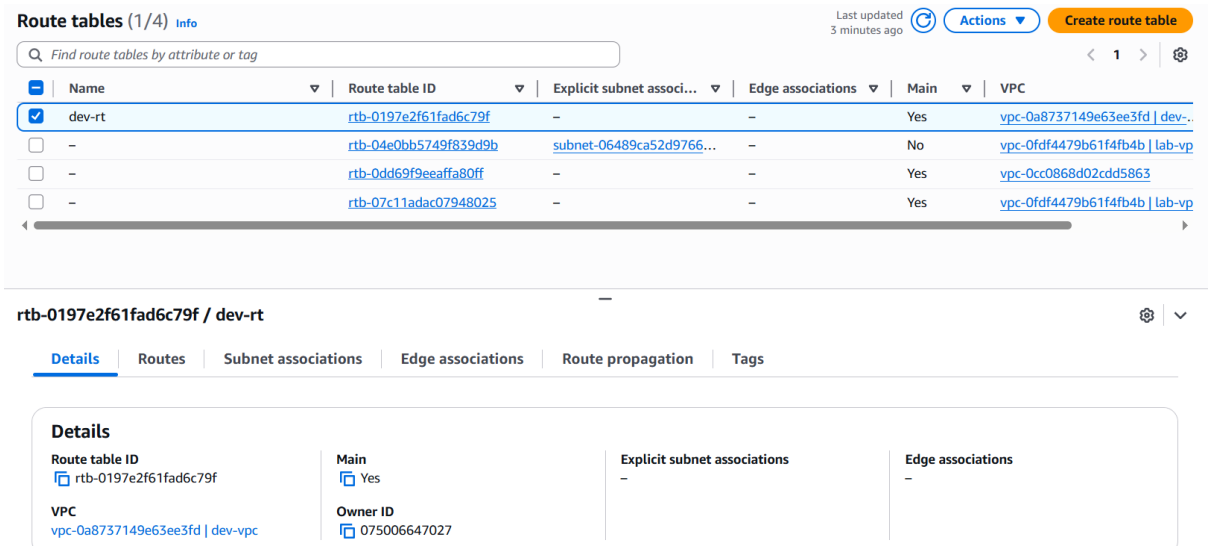


q2_console_subnet

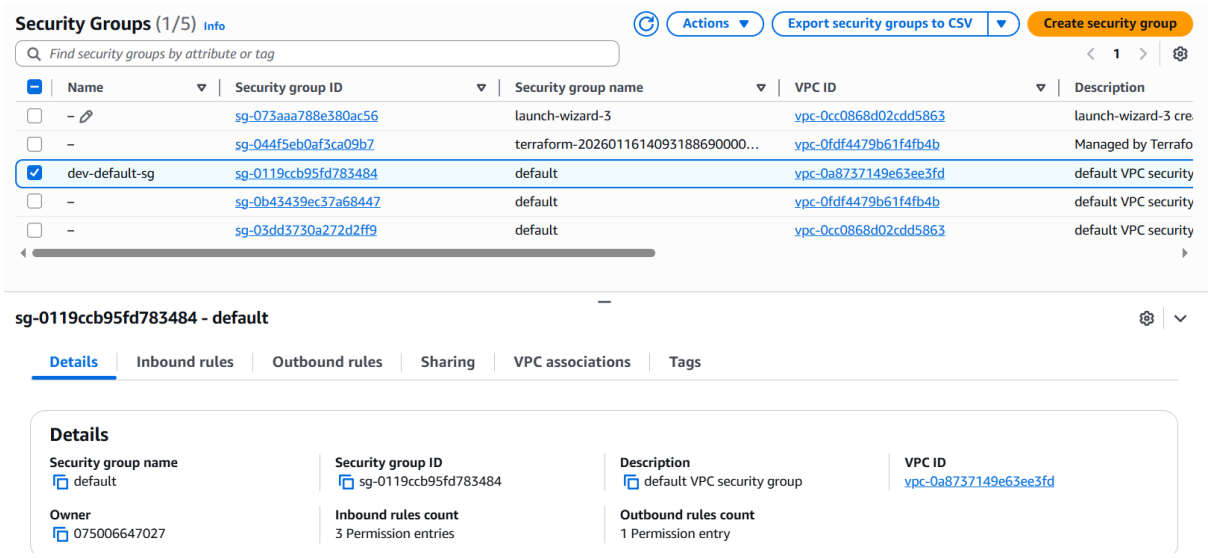
q2_console_igw



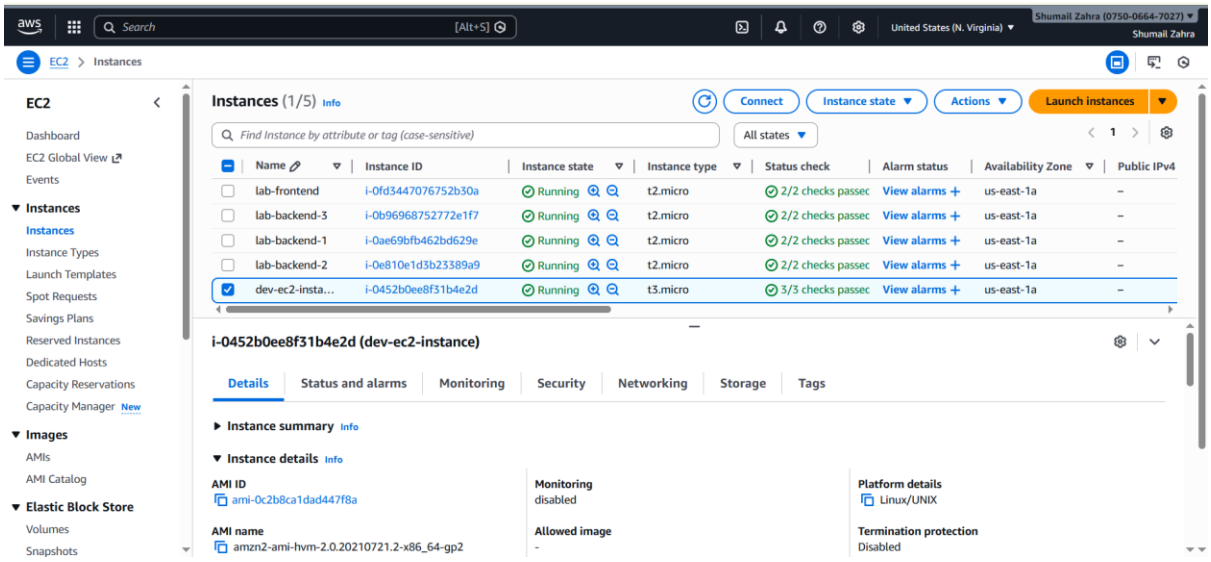
q2_console_route_table



q2_console_sg



q2_console_ec2



q2_https_browser

This is Shumail's Terraform environment

Q3

q3_hosts

```
GNU nano 7.2                                    hosts *
[ec2]
3.238.32.182

[ec2:vars]
ansible_user=ec2-user
ansible_ssh_private_key_file=~/.ssh/id_ed25519
ansible_ssh_common_args='-o StrictHostKeyChecking=no'
```

q3_ansible_cfg

```
[defaults]
host_key_checking = False
inventory = ./hosts
remote_python_interpreter = /usr/bin/python3
~
~
~
~
~
~
~
~
```

q3_playbook

```yaml
---
- name: EC2 Apache Web Server Setup
  hosts: ec2
  become: true
  tasks:

    - name: Update all packages
      amazon.aws.yum:
        name: "*"
        state: latest

    - name: Stop Nginx if running
      systemd:
        name: nginx
        state: stopped
        enabled: no
      ignore_errors: yes

    - name: Remove Nginx if installed
      yum:
        name: nginx
        state: absent
      ignore_errors: yes

    - name: Install Apache HTTPD
      yum:
        name: httpd
        state: present

    - name: Start and enable Apache HTTPD
      systemd:
        name: httpd
        state: started
        enabled: yes
```

q3_http_browser

This is Shumail's Terraform environment

Cleanup (ungraded)

cleanup_terraform_destroy

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $  terraform destroy -auto-approve
      - tags_all                              = {
          - "Name" = "dev-vpc"
        } -> null
        # (4 unchanged attributes hidden)
    }

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

Changes to Outputs:
  - ec2_public_ip = "3.238.32.182" -> null
aws_default_route_table.myapp_rt: Destroying... [id=rtb-0197e2f61fad6c79f]
aws_instance.myapp_ec2: Destroying... [id=i-0452b0ee8f31b4e2d]
aws_default_route_table.myapp_rt: Destruction complete after 0s
aws_internet_gateway.myapp_igw: Destroying... [id=igw-022dc4343a629921f]
aws_instance.myapp_ec2: Still destroying... [id=i-0452b0ee8f31b4e2d, 00m10s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-022dc4343a629921f, 00m10s elapsed]
aws_instance.myapp_ec2: Still destroying... [id=i-0452b0ee8f31b4e2d, 00m20s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-022dc4343a629921f, 00m20s elapsed]
aws_instance.myapp_ec2: Still destroying... [id=i-0452b0ee8f31b4e2d, 00m30s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-022dc4343a629921f, 00m30s elapsed]
aws_instance.myapp_ec2: Still destroying... [id=i-0452b0ee8f31b4e2d, 00m40s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-022dc4343a629921f, 00m40s elapsed]
aws_internet_gateway.myapp_igw: Destruction complete after 43s
aws_instance.myapp_ec2: Destruction complete after 44s
aws_key_pair.serverkey: Destroying... [id=serverkey]
aws_subnet.myapp_subnet: Destroying... [id=subnet-03c2018a212fb80f7]
aws_default_security_group.default_sg: Destroying... [id=sg-0119ccb95fd783484]
aws_default_security_group.default_sg: Destruction complete after 0s
aws_key_pair.serverkey: Destruction complete after 1s
aws_subnet.myapp_subnet: Destruction complete after 2s
aws_vpc.myapp_vpc: Destroying... [id=vpc-0a8737149e63ee3fd]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 7 destroyed.
@23-22411-061-rgb →/workspaces/Lab_exam (main) $
```

cleanup_ec2_console