
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": "AIDARC5V6TLZYZZZYNN4",
    "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": "AIDARC5V6TLZYTZZYNN4",
    "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) $

```

q1_group_membership

```

@23-22411-061-rgb →/workspaces/Lab_exam (main) $ aws iam get-group --group-name SoftwareEngineering
{
  "Users": [
    {
      "Path": "/",
      "UserName": "Shumail",
      "UserId": "AIDARC5V6TLZYTZZYNN4",
      "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=='AdministratorAccess']"
[
  {
    "PolicyName": "AdministratorAccess",
    "PolicyId": "ANPAIWMBCSKIEE64ZLYK",
    "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": "ANPAIWMBCSKIEE64ZLYK",
    "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 SoftwareEngineering
{
  "AttachedPolicies": [
    {
      "PolicyName": "AdministratorAccess",
      "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
    }
  ]
}
```

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $
```

q1_console_group

The screenshot shows the AWS IAM console interface. The main content area displays 'User groups (1) Info'. A table lists the user groups, with one group named 'SoftwareEngineering' having 1 user and 'Defined' permissions. The left sidebar shows the navigation menu with 'User groups' selected under the 'Access Management' section.

q1_console_user_in_group

Identity and Access Management (IAM) > Users

Search IAM

Dashboard

Access Management

- User groups
- Users**
- Roles
- Policies
- Identity providers
- Account settings
- Root access management
- Temporary delegation requests
- [New](#)

Access reports

- Access Analyzer
- Resource analysis [New](#)

Users (2) [Info](#)

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

Search

<input type="checkbox"/>	User name	Path	Groups	Last activity	MFA	Password age	Console last sign-in	Acc
<input type="checkbox"/>	ec2	/	0	5 minutes ago	-	-	-	Act
<input type="checkbox"/>	Shumail	/	1	-	-	-	-	-

q1_console_group_policy

Identity and Access Management (IAM) > User groups > SoftwareEngineering

Search IAM

Dashboard

Access Management

- User groups**
- Users
- Roles
- Policies
- Identity providers
- Account settings
- Root access management
- Temporary delegation requests
- [New](#)

Access reports

- Access Analyzer
- Resource analysis [New](#)

SoftwareEngineering [Info](#)

[Delete](#)

[Edit](#)

Summary

User group name: SoftwareEngineering

Creation time: January 19, 2026, 12:45 (UTC+05:00)

ARN: [arn:aws:iam::075006647027:group/SoftwareEngineering](#)

Users (1) **Permissions** Access Advisor

Permissions policies (1) [Info](#)

You can attach up to 10 managed policies.

Filter by Type: All types

<input type="checkbox"/>	Policy name	Type	Attached entities
<input type="checkbox"/>	AdministratorAccess	AWS managed - job function	2

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.id]
  availability_zone = var.availability_zone
  associate_public_ip_address = true
  key_name         = aws_key_pair.serverkey.key_name
}

```

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

```

@23-22411-061-rgb →/workspaces/Lab_exam (main) $ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Finding latest version of hashicorp/http...
- Installing hashicorp/aws v6.28.0...
- Installed hashicorp/aws v6.28.0 (signed by HashiCorp)
- Installing hashicorp/http v3.5.0...
- Installed hashicorp/http v3.5.0 (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.

```

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"
```

```
@23-22411-061-rgb →/workspaces/Lab_exam (main) $
```

q2_console_vpc

your VPCs

VPCsVPC encryption controls

Your VPCs (3) Info

Last updated less than a minute ago

Actions

Create VPC

Find VPCs by attribute or tag

<input type="checkbox"/>	Name	VPC ID	State	Encryption c...	Encryption control ...	Block Public...	IPv...
<input type="checkbox"/>	lab-vpc	vpc-0fdf4479b61f4fb4b	Available	-	-	Off	10.0.1.0/24
<input type="checkbox"/>	dev-vpc	vpc-0a8737149e63ee3fd	Available	-	-	Off	10.0.10.0/24
<input type="checkbox"/>	-	vpc-0cc0868d02cdd5863	Available	-	-	Off	172.31.0.0/20

Select a VPC above

Your VPCs (1/3) Info

Last updated 1 minute ago

Actions

Create VPC

Find VPCs by attribute or tag

<input type="checkbox"/>	Name	VPC ID	State	Encryption c...	Encryption control ...	Block Public...	IPv...
<input type="checkbox"/>	lab-vpc	vpc-0fdf4479b61f4fb4b	Available	-	-	Off	10.0.1.0/24
<input checked="" type="checkbox"/>	dev-vpc	vpc-0a8737149e63ee3fd	Available	-	-	Off	10.0.10.0/24
<input type="checkbox"/>	-	vpc-0cc0868d02cdd5863	Available	-	-	Off	172.31.0.0/20

vpc-0a8737149e63ee3fd / dev-vpc

Details

VPC ID

vpc-0a8737149e63ee3fd

DNS resolution

Enabled

Main network ACL

acl-09b51562f57ce2c86

IPv6 CIDR (Network border group)

State

Available

Tenancy

default

Default VPC

No

Network Address Usage metrics

Block Public Access

Off

DHCP option set

dopt-03c04818ffb45a2df

IPv4 CIDR

10.0.0.0/16

Route 53 Resolver DNS Firewall rule

DNS hostnames

Disabled

Main route table

rtb-0197e2f61fad6c79f

IPv6 pool

-

Owner ID

q2_console_subnet

<input type="checkbox"/>	-	subnet-06489ca52d97665dc	Available	vpc-0fdf4479b61f4fb4b lab-vpc	Off	10.0.1.0/24
<input type="checkbox"/>	-	subnet-05c9ebf1230d6a267	Available	vpc-0cc0868d02cdd5863	Off	172.31.80.0/20
<input checked="" type="checkbox"/>	dev-subnet-1	subnet-03c2018a212fb80f7	Available	vpc-0a8737149e63ee3fd dev-...	Off	10.0.10.0/24
<input type="checkbox"/>	-	subnet-0ad57ddb11a1febc	Available	vpc-0cc0868d02cdd5863	Off	172.31.32.0/20
<input type="checkbox"/>	-	subnet-0db578dbff650c91c	Available	vpc-0cc0868d02cdd5863	Off	172.31.16.0/20
<input type="checkbox"/>	-	subnet-03ba52752c6cd3922	Available	vpc-0cc0868d02cdd5863	Off	172.31.0.0/20

subnet-03c2018a212fb80f7 / dev-subnet-1

Details

Subnet ID

subnet-03c2018a212fb80f7

IPv4 CIDR

10.0.10.0/24

Availability Zone

use1-az1 (us-east-1a)

Network ACL

acl-09b51562f57ce2c86

Subnet ARN

arn:aws:ec2:us-east-1:075006647027:subnet/subnet-03c2018a212fb80f7

Available IPv4 addresses

250

Network border group

us-east-1

Default subnet

No

State

Available

IPv6 CIDR

-

VPC

vpc-0a8737149e63ee3fd | dev-vpc

Auto-assign public IPv4 address

Yes

Block Public Access

Off

IPv6 CIDR association ID

-

Route table

rtb-0197e2f61fad6c79f | dev-rt

Auto-assign IPv6 address

No

Mobile App

© 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

q2_console_igw

Internet gateways (1/3) Info

Find internet gateways by attribute or tag

Actions

Create internet gateway

<input type="checkbox"/>	Name	Internet gateway ID	State	VPC ID	Owner
<input type="checkbox"/>	-	igw-01c4c2289ed245c83	Attached	vpc-0cc0868d02cdd5863	075006647027
<input checked="" type="checkbox"/>	dev-igw	igw-022dc4343a629921f	Attached	vpc-0a8737149e63ee3fd dev-vpc	075006647027
<input type="checkbox"/>	-	igw-0a134d50262aa1a91	Attached	vpc-0fdf4479b61f4fb4b lab-vpc	075006647027

igw-022dc4343a629921f / dev-igw

Details Tags

Details

Internet gateway ID
igw-022dc4343a629921f

State
Attached

VPC ID
vpc-0a8737149e63ee3fd | dev-vpc

Owner
075006647027

q2_console_route_table

Route tables (1/4) Info

Find route tables by attribute or tag

Last updated
3 minutes ago

Actions

Create route table

<input type="checkbox"/>	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
<input checked="" type="checkbox"/>	dev-rt	rtb-0197e2f61fad6c79f	-	-	Yes	vpc-0a8737149e63ee3fd dev-..
<input type="checkbox"/>	-	rtb-04e0bb5749f839d9b	subnet-06489ca52d9766...	-	No	vpc-0fdf4479b61f4fb4b lab-vp
<input type="checkbox"/>	-	rtb-0dd69f9eeaffa80ff	-	-	Yes	vpc-0cc0868d02cdd5863
<input type="checkbox"/>	-	rtb-07c11adac07948025	-	-	Yes	vpc-0fdf4479b61f4fb4b lab-vp

rtb-0197e2f61fad6c79f / dev-rt

Details Routes Subnet associations Edge associations Route propagation Tags

Details

Route table ID
rtb-0197e2f61fad6c79f

Main
Yes

Explicit subnet associations
-

Edge associations
-

VPC
vpc-0a8737149e63ee3fd | dev-vpc

Owner ID
075006647027

q2_console_sg

Security Groups (1/5) Info

Find security groups by attribute or tag

Actions

Export security groups to CSV

Create security group

<input type="checkbox"/>	Name	Security group ID	Security group name	VPC ID	Description
<input type="checkbox"/>	-	sg-073aaa788e380ac56	launch-wizard-3	vpc-0cc0868d02cdd5863	launch-wizard-3 cre
<input type="checkbox"/>	-	sg-044f5eb0af3ca09b7	terraform-2026011614093188690000...	vpc-0fdf4479b61f4fb4b	Managed by Terrafo
<input checked="" type="checkbox"/>	dev-default-sg	sg-0119ccb95fd783484	default	vpc-0a8737149e63ee3fd	default VPC security
<input type="checkbox"/>	-	sg-0b43439ec37a68447	default	vpc-0fdf4479b61f4fb4b	default VPC security
<input type="checkbox"/>	-	sg-03dd3730a272d2ff9	default	vpc-0cc0868d02cdd5863	default VPC security

sg-0119ccb95fd783484 - default

Details Inbound rules Outbound rules Sharing VPC associations Tags

Details

Security group name
default

Security group ID
sg-0119ccb95fd783484

Description
default VPC security group

VPC ID
vpc-0a8737149e63ee3fd

Owner
075006647027

Inbound rules count
3 Permission entries

Outbound rules count
1 Permission entry

q2_console_ec2

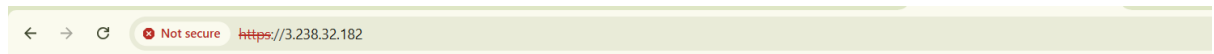
The screenshot shows the AWS Management Console for the 'United States (N. Virginia)' region. The left sidebar shows the navigation menu with 'EC2' selected. The main content area displays a list of EC2 instances. The instance 'dev-ec2-instance' (ID: i-0452b0ee8f31b4e2d) is selected. Below the list, the details for this instance are shown, including the AMI ID (ami-0c2b8ca1dad447f8a), AMI name (amzn2-ami-hvm-2.0.20210721.2-x86_64-gp2), and monitoring status (disabled).

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
lab-frontend	i-0fd3447076752b30a	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
lab-backend-3	i-0b96968752772e1f7	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
lab-backend-1	i-0ae69bfb462bd629e	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
lab-backend-2	i-0e810e1d3b23389a9	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
dev-ec2-instance	i-0452b0ee8f31b4e2d	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	-

Instance details for i-0452b0ee8f31b4e2d (dev-ec2-instance)

- AMI ID:** ami-0c2b8ca1dad447f8a
- AMI name:** amzn2-ami-hvm-2.0.20210721.2-x86_64-gp2
- Monitoring:** disabled
- Platform details:** Linux/UNIX
- Termination protection:** Disabled

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


```
--
- 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

← → ↻ Not secure https://3.238.32.182

This is Shumail's Terraform environment

Cleanup (ungraded)

cleanup_terraform_destroy

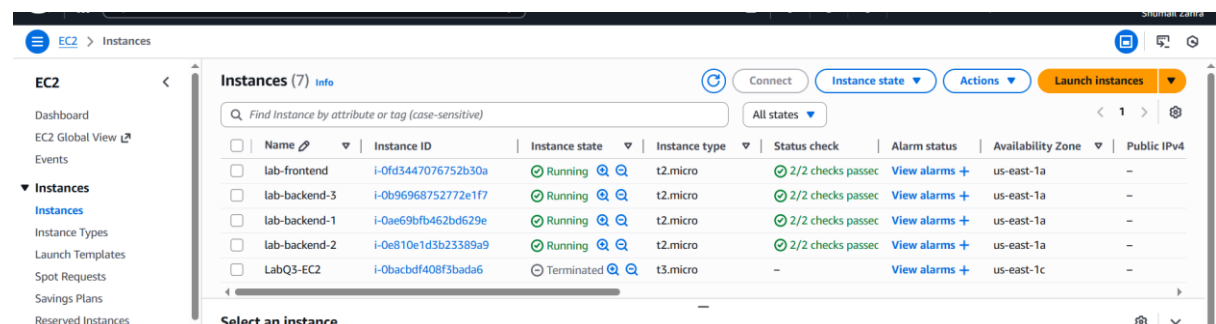
```
@23-22411-061-rgb → /workspaces/Lab_exam (main) $ terraform destroy -auto-approve
- tags_all = {
  - "Name" = "dev-vpc"
} -> null
# (4 unchanged attributes hidden)
}

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



	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
<input type="checkbox"/>	lab-frontend	i-0fd3447076752b30a	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
<input type="checkbox"/>	lab-backend-3	i-0b96968752772e1f7	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
<input type="checkbox"/>	lab-backend-1	i-0ae69bfb462bd629e	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
<input type="checkbox"/>	lab-backend-2	i-0e810e1d3b23389a9	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
<input type="checkbox"/>	LabQ3-EC2	i-0bacbd408f3bada6	Terminated	t3.micro	-	View alarms +	us-east-1c	-