Module 9: Terraform Overview

~ Abhijit Zende

Provision resources on AWS using Terraform. Resources to provision:

1. EC2 Instance
2. VPC
3. Subnet
4. Internet Gateway
5. Route-Table
6. RouteTable Association
7. Security Group

 # Detailed notes, Guide & code on my GitHub #

https://github.com/Abhiz2411/terraform-ninja-devops-beginner-guide

1. L1 - Provision AWS EC2 Instance along with VPC, Subnet, Internet Gateway, Route-Table, Route Table Association, Security Group

Ans.

(*** Note: Screen shots attached to end of each question ***)

Prerequisites:

1. **Install VS Code:** Download and Install VS Code IDE. Install Terraform extension on VS Code.
2. **Install Terraform**: Ensure Terraform is installed and configured with your AWS credentials.
3. **AWS Account**: Ensure you have an AWS **IAM** account with the necessary permissions to create resources.

Steps:

1. **Create a main.tf file to define all the resources.**
   a. The code will be in below format:

```
# Configure the AWS Provider
provider "aws" {
  region = "ap-south-1" # Specify your desired region
}

# 1. Create VPC
resource "aws_vpc" "main" {
  # CIDR block for the VPC - This will give us 65,536 private IP addresses
  cidr_block = "10.0.0.0/16"

  # Enable DNS hostnames for instances in the VPC
  enable_dns_hostnames = true

  # Enable DNS support in the VPC
  enable_dns_support = true
```

```
  tags = {
    Name = "main-vpc"
  }
}

# 2. Create Public Subnet
resource "aws_subnet" "public" {
  # VPC ID where subnet will be created
  vpc_id = aws_vpc.main.id

  # CIDR block for the subnet - This gives us 256 IP addresses
  cidr_block = "10.0.1.0/24"

  # AZ where the subnet will be created
  availability_zone = "ap-south-1a"

  # Enable auto-assign public IP addresses for instances in this subnet
  map_public_ip_on_launch = true

  tags = {
    Name = "public-subnet"
  }
}

# 3. Create Internet Gateway
# This allows communication between our VPC and the internet
resource "aws_internet_gateway" "main" {
  # Attach the IGW to our VPC
  vpc_id = aws_vpc.main.id

  tags = {
    Name = "main-igw"
  }
}

# 4. Create Route Table
# This will control where network traffic is directed
resource "aws_route_table" "public" {
```

```
  vpc_id = aws_vpc.main.id

  # Route all IPv4 traffic (0.0.0.0/0) to the internet gateway
  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.main.id
  }

  tags = {
    Name = "public-rt"
  }
}

# 5. Create Route Table Association
# This associates our public subnet with our public route table
resource "aws_route_table_association" "public" {
  subnet_id      = aws_subnet.public.id
  route_table_id = aws_route_table.public.id
}

# 6. Create Security Group
# This acts as a virtual firewall for our EC2 instance
resource "aws_security_group" "allow_ssh_http" {
  name        = "allow_ssh_http"
  description = "Allow SSH and HTTP inbound traffic"
  vpc_id      = aws_vpc.main.id

  # Allow incoming SSH traffic (port 22) from any IP
  ingress {
    description = "SSH from anywhere"
    from_port   = 22
    to_port     = 22
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  # Allow incoming HTTP traffic (port 80) from any IP
  ingress {
```

```
    description = "HTTP from anywhere"
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  # Allow all outbound traffic
  egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }

  tags = {
    Name = "allow_ssh_http"
  }
}

# 7. Get the latest Amazon Linux 2 AMI ID
data "aws_ami" "amazon_linux_2" {
  most_recent = true
  owners      = ["amazon"]

  filter {
    name   = "name"
    values = ["amzn2-ami-hvm-*-x86_64-gp2"]
  }

  filter {
    name   = "virtualization-type"
    values = ["hvm"]
  }
}

# 8. Create EC2 Instance
resource "aws_instance" "web_server" {
```

```
  # Use the AMI ID we fetched
  ami = data.aws_ami.amazon_linux_2.id

  # Use t2.micro instance type (free tier eligible)
  instance_type = "t2.micro"

  # Launch in our public subnet
  subnet_id = aws_subnet.public.id

  # Use the security group we created
  vpc_security_group_ids = [aws_security_group.allow_ssh_http.id]

  # Name the instance
  tags = {
    Name = "web-server"
  }

  # User data script to install and start Apache web server
  user_data = <<-EOF
              #!/bin/bash
              yum update -y
              yum install -y httpd
              systemctl start httpd
              systemctl enable httpd
              echo "<h1>Hello from Terraform</h1>" >
/var/www/html/index.html
              EOF
  }

  # Output the public IP of the EC2 instance
  output "public_ip" {
    value       = aws_instance.web_server.public_ip
    description = "Public IP address of the web server"
  }
  ```
```

b. **Provider:** We specify the AWS region to use.
c. **VPC:** Creates a VPC with a CIDR block 10.0.0.0/16.
d. **Subnet:** Creates a public subnet with a CIDR block 10.0.1.0/24.
```

e. **Internet Gateway:** Attaches an internet gateway to the VPC to enable internet access.
f. **Route Table and Route:** Creates a route table and a route that directs internet-bound traffic (0.0.0.0/0) through the internet gateway.
g. **Route Table Association:** Associates the route table with the subnet to apply the routing rules.
h. **Security Group:** Defines a security group allowing SSH (port 22) and HTTP (port 80) inbound access, and all outbound traffic.
i. **EC2 Instance:** Launches an EC2 instance with a specified AMI (Amazon Machine Image), instance type, and security group, and associates a public IP.

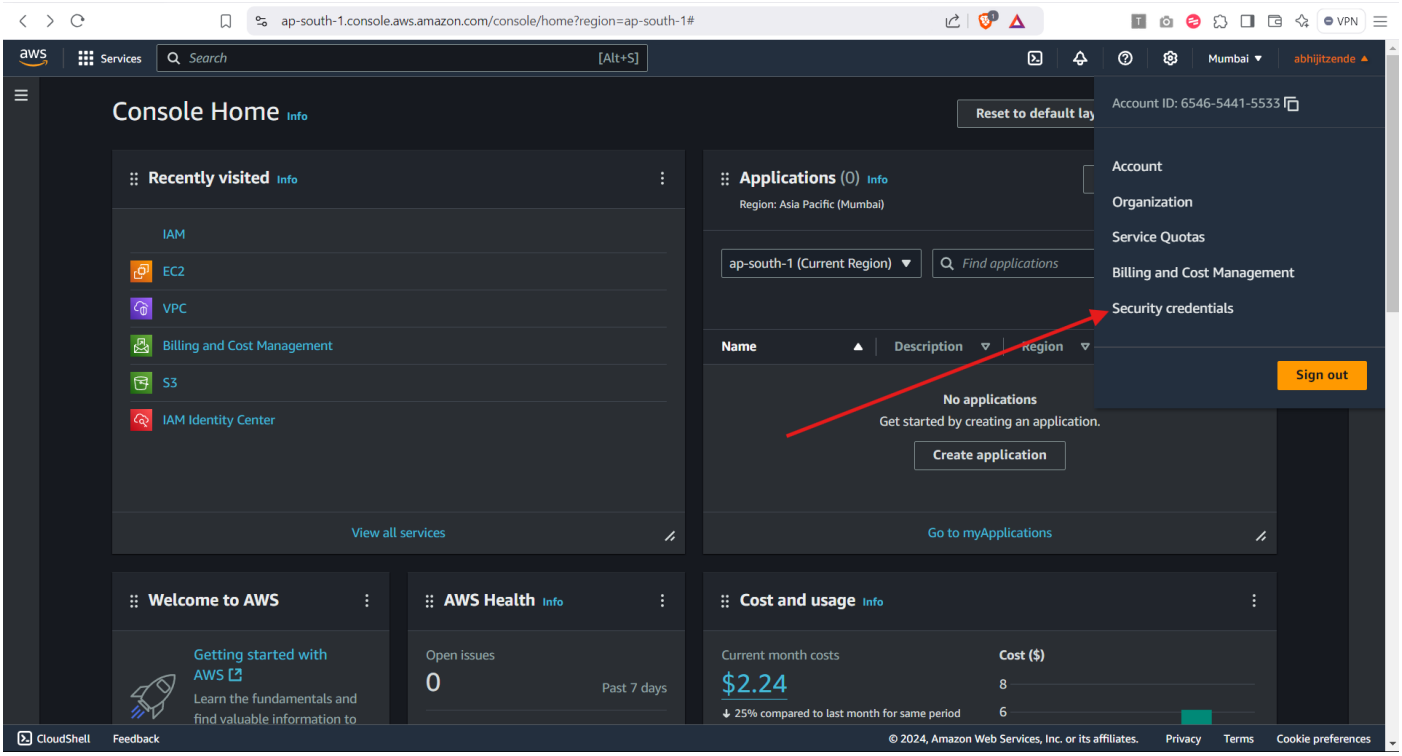
2. **Initialize Terraform**: In your terminal, navigate to the directory where the main.tf file is located and run:
   `terraform init`

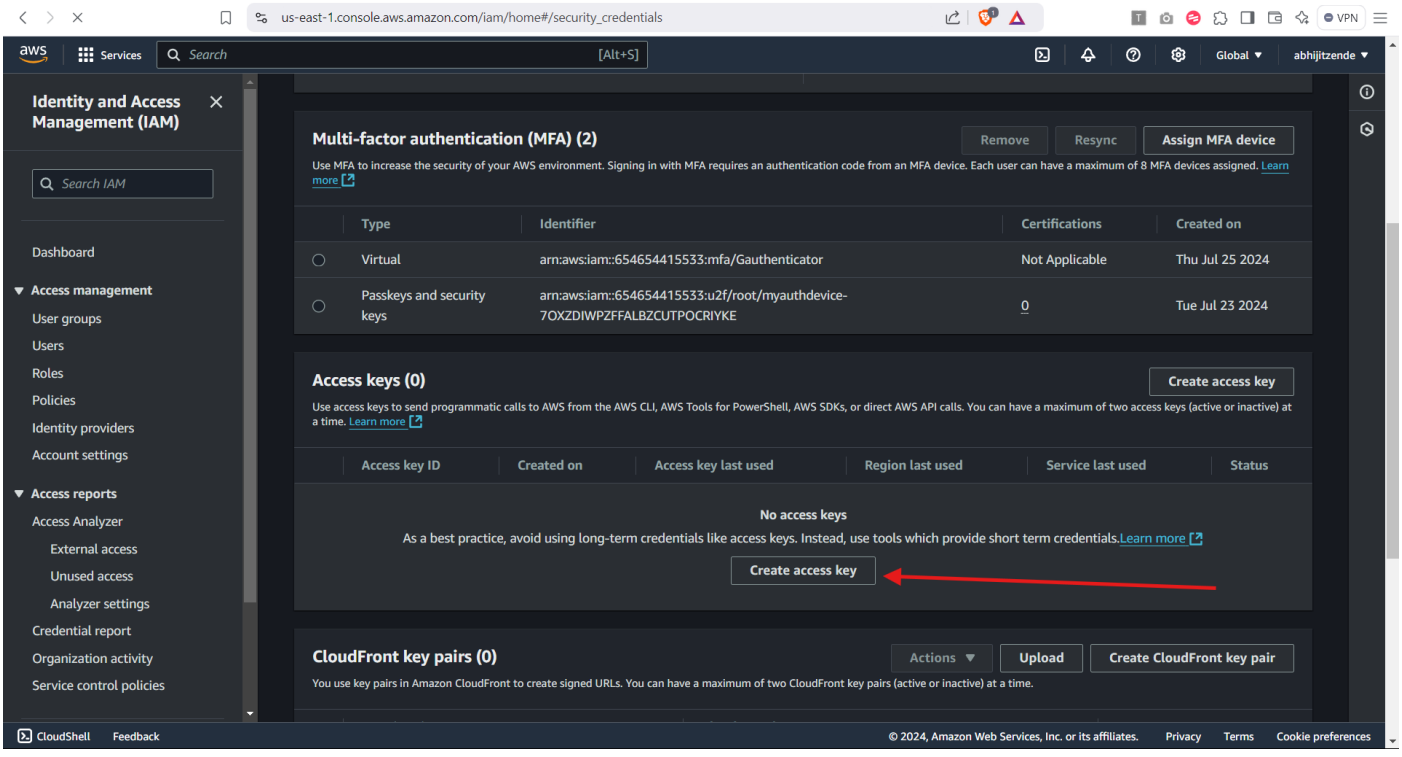
3. Plan the Deployment: Terraform will show you the actions it plans to take.
   `terraform plan`

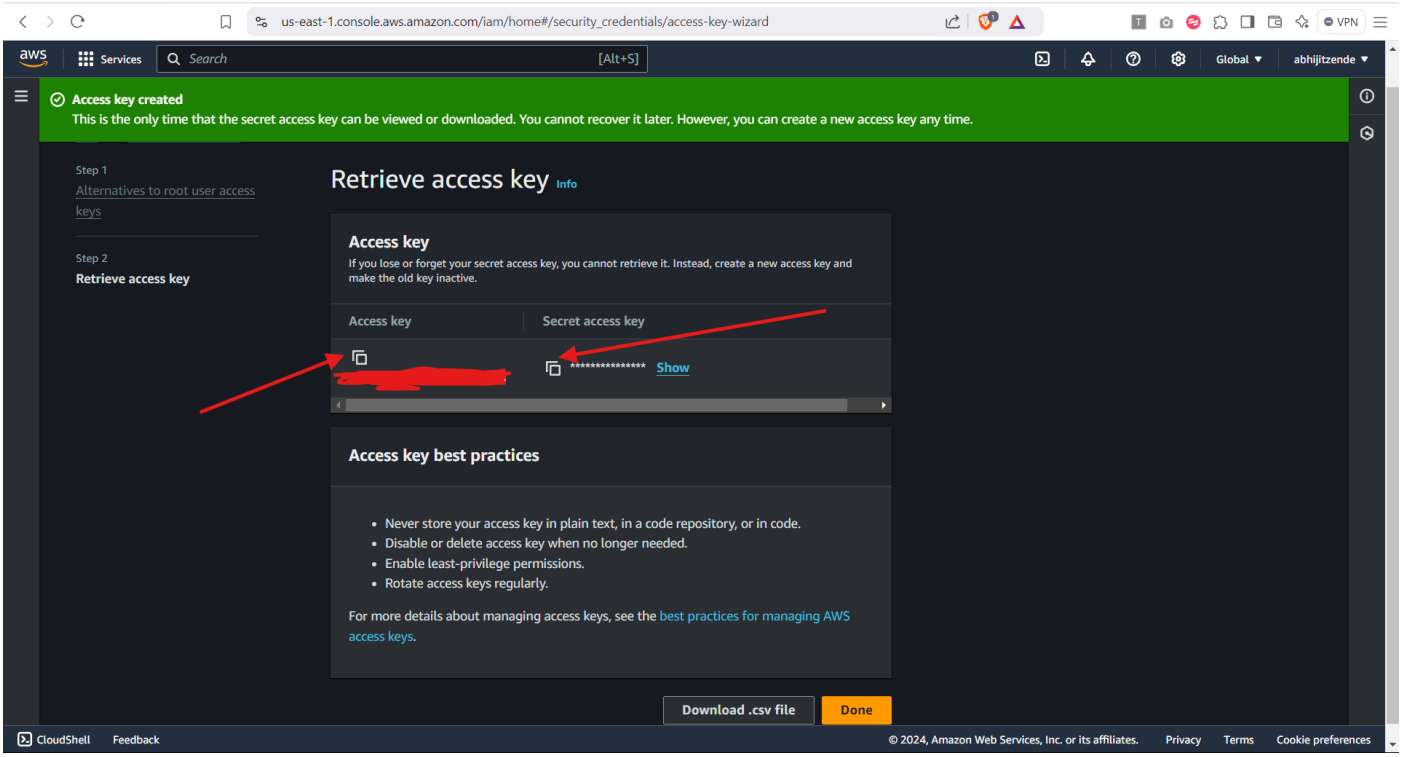
4. Apply the Configuration: To create the resources, run:
   `terraform apply`

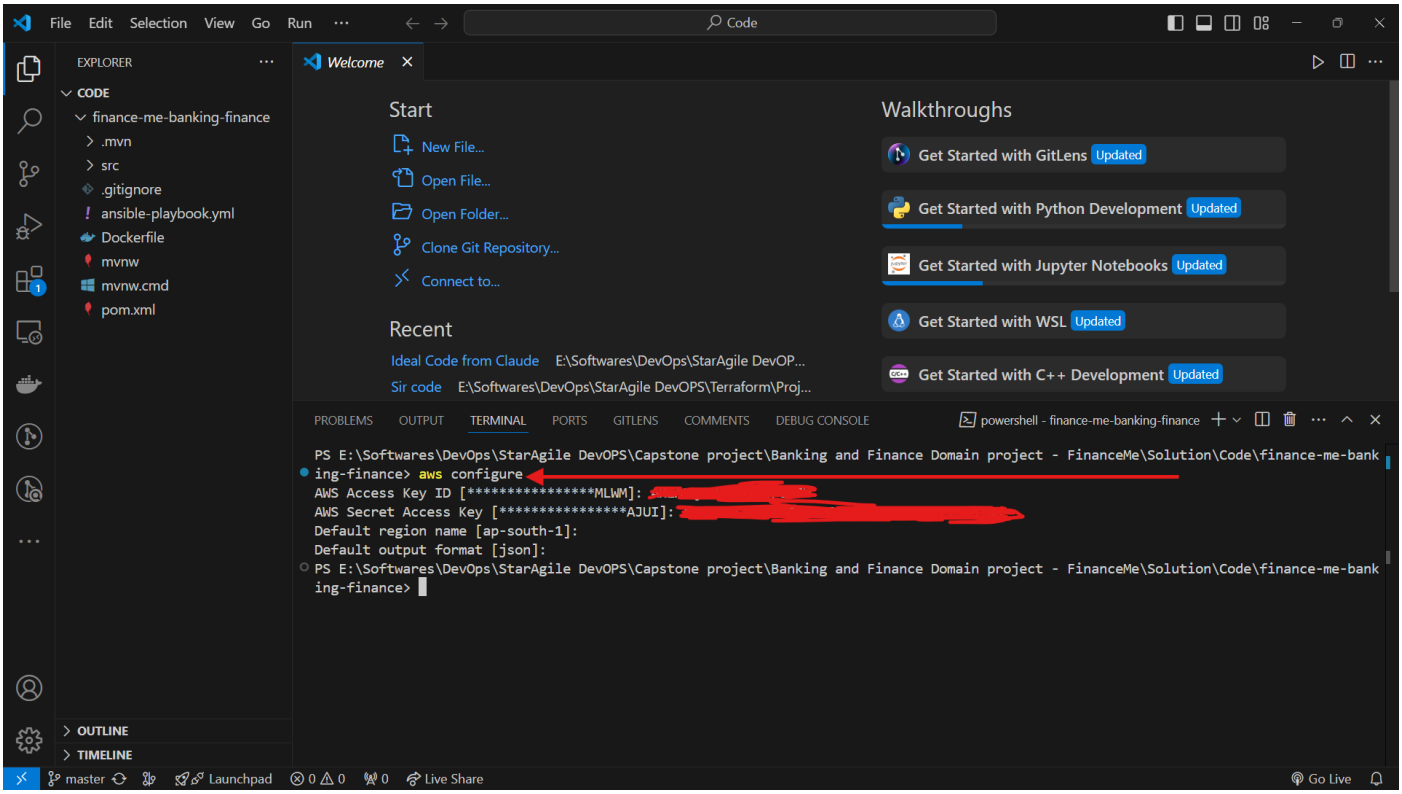
5. **Verify the Resources**: After successful deployment, you can log in to the AWS Management Console to verify that the VPC, subnet, security group, EC2 instance, and other resources have been created.

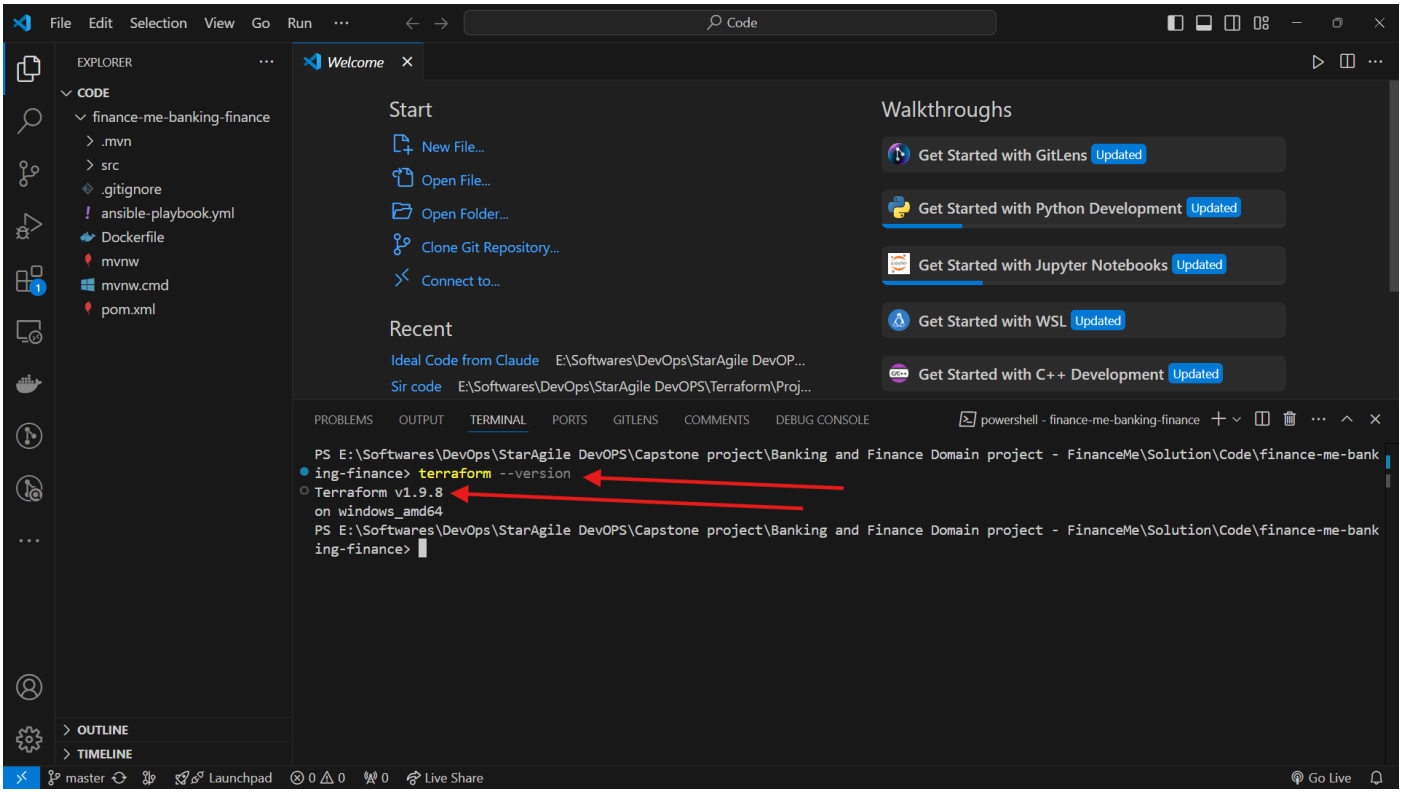
6. **Clean Up (Optional)**: To destroy all the resources created by Terraform, run:
   `terraform destroy`

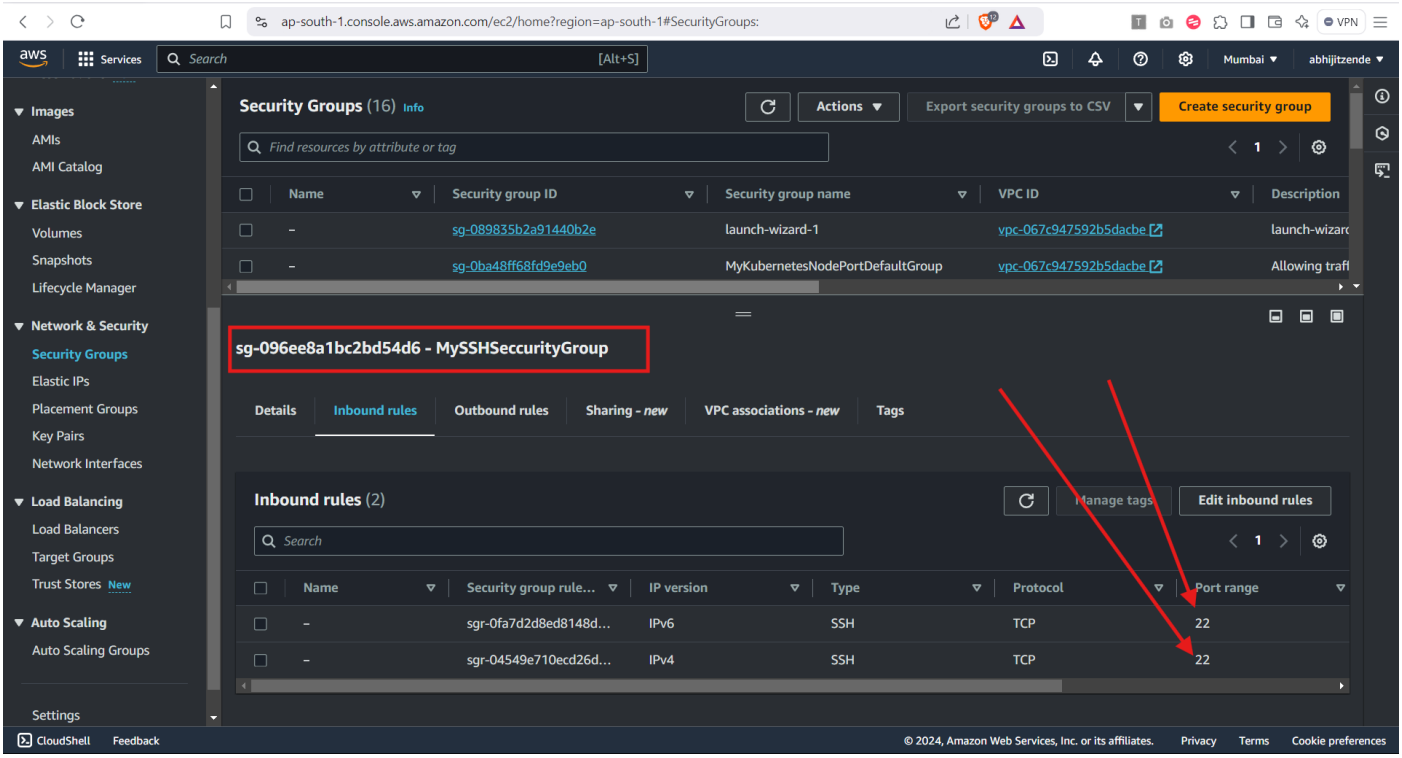

ap-south-1.console.aws.amazon.com/console/home?region=ap-south-1#
Services
Search
[Alt+S]
Mumbai
abhijitzende
Console Home Info
Reset to default lay
Account ID: 6546-5441-5533
Account
Organization
Service Quotas
Billing and Cost Management
Security credentials
Sign out
Recently visited Info
IAM
EC2
VPC
Billing and Cost Management
S3
IAM Identity Center
View all services
Applications (0) Info
Region: Asia Pacific (Mumbai)
ap-south-1 (Current Region)
Find applications
Name
Description
Region
No applications
Get started by creating an application.
Create application
Go to myApplications
Welcome to AWS
Getting started with AWS
Learn the fundamentals and find valuable information to
AWS Health Info
Open issues
0
Past 7 days
Cost and usage Info
Current month costs
$2.24
25% compared to last month for same period
Cost ($)
8
6
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

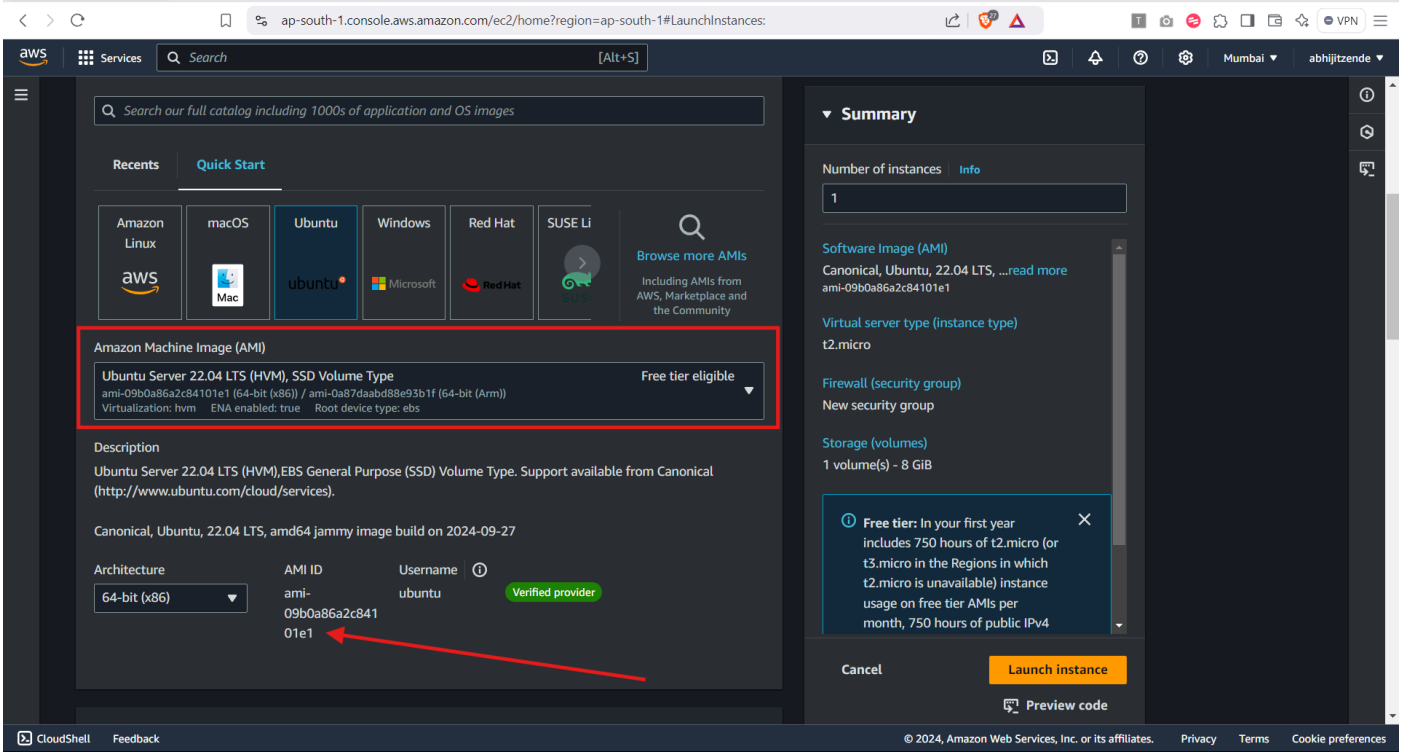

us-east-1.console.aws.amazon.com/iam/home#/security_credentials
Services
Search
[Alt+S]
Global
abhijitzende
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Credential report
Organization activity
Service control policies
Multi-factor authentication (MFA) (2)
Remove
Resync
Assign MFA device
Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more
Type
Identifier
Certifications
Created on
Virtual
arn:aws:iam::654654415533:mfa/Gauthenticator
Not Applicable
Thu Jul 25 2024
Passkeys and security keys
arn:aws:iam::654654415533:u2f/root/myauthdevice-7OXZDIWPZFFALBZCUTPOCRIYKE
0
Tue Jul 23 2024
Access keys (0)
Create access key
Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. Learn more
Access key ID
Created on
Access key last used
Region last used
Service last used
Status
No access keys
As a best practice, avoid using long-term credentials like access keys. Instead, use tools which provide short term credentials. Learn more
Create access key
CloudFront key pairs (0)
Actions
Upload
Create CloudFront key pair
You use key pairs in Amazon CloudFront to create signed URLs. You can have a maximum of two CloudFront key pairs (active or inactive) at a time.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

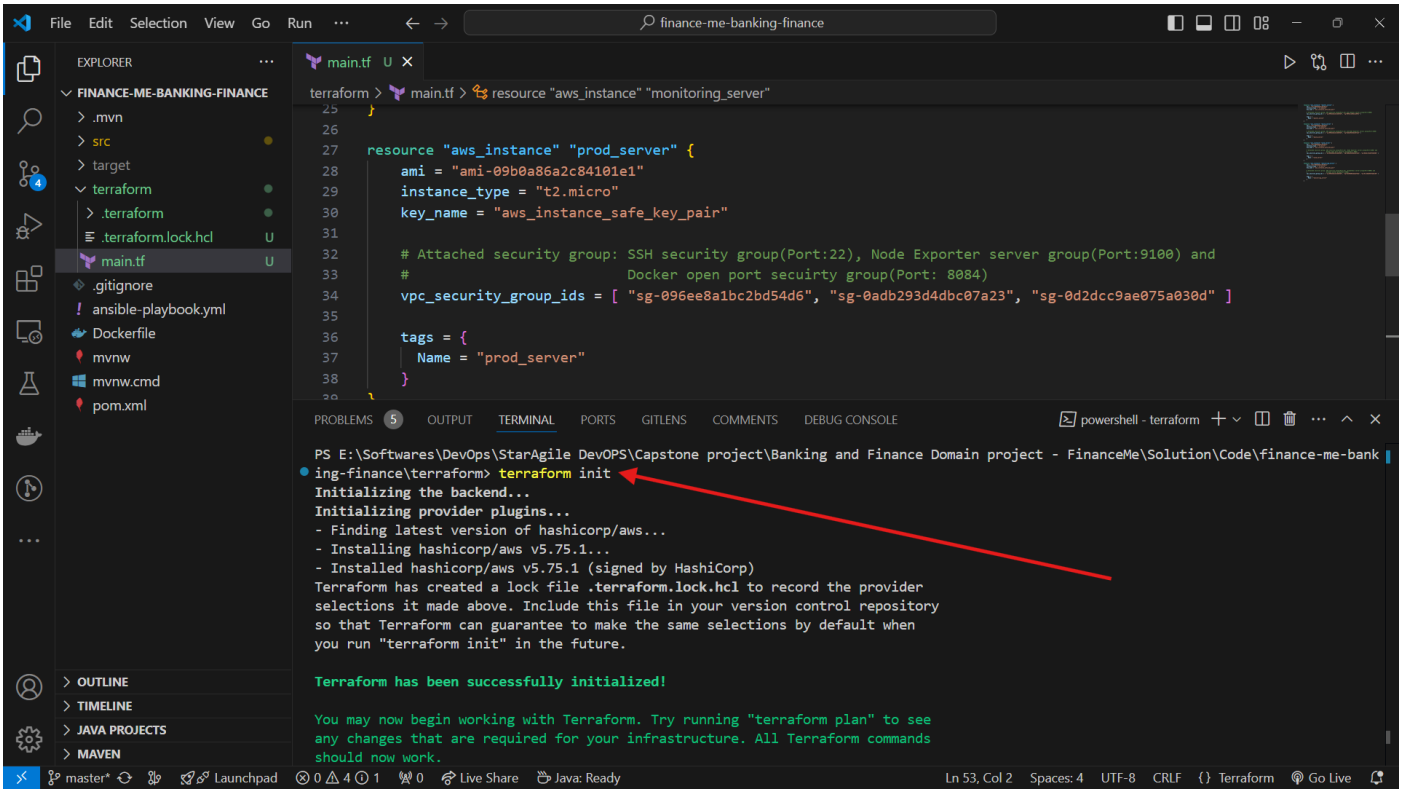

us-east-1.console.aws.amazon.com/iam/home#/security_credentials/access-key-wizard
aws
Services
Search
[Alt+S]
Global
abhijitzende
Access key created
This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.
Step 1
Alternatives to root user access keys
Step 2
Retrieve access key
Retrieve access key Info
Access key
If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.
Access key
Secret access key
********** Show
Access key best practices
Never store your access key in plain text, in a code repository, or in code.
Disable or delete access key when no longer needed.
Enable least-privilege permissions.
Rotate access keys regularly.
For more details about managing access keys, see the best practices for managing AWS access keys.
Download .csv file
Done
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

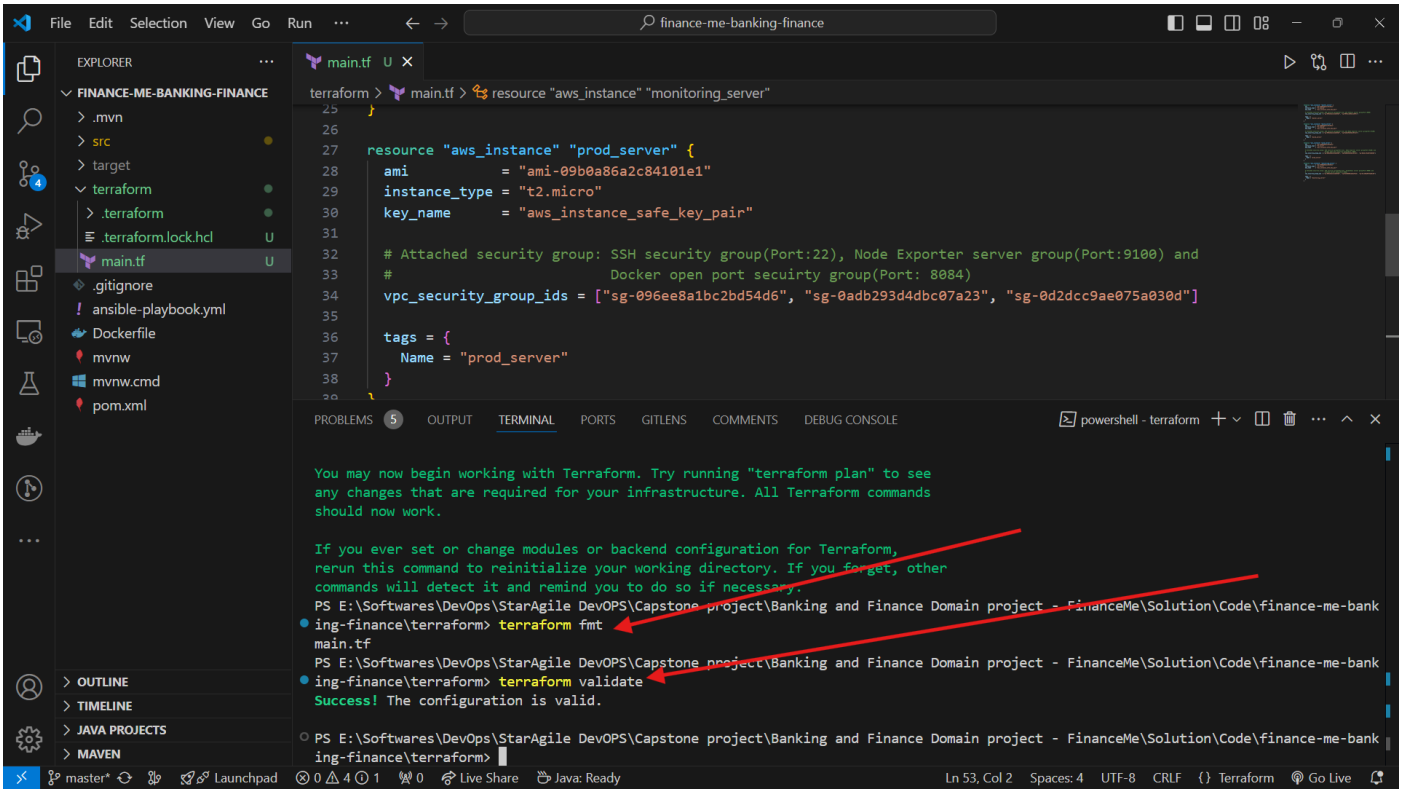

File Edit Selection View Go Run
Code
EXPLORER
CODE
finance-me-banking-finance
.mvn
src
.gitignore
ansible-playbook.yml
Dockerfile
mvnw
mvnw.cmd
pom.xml
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
Recent
Ideal Code from Claude E:\Softwares\DevOps\StarAgile DevOP...
Sir code E:\Softwares\DevOps\StarAgile DevOPS\Terraform\Proj...
Walkthroughs
Get Started with GitLens Updated
Get Started with Python Development Updated
Get Started with Jupyter Notebooks Updated
Get Started with WSL Updated
Get Started with C++ Development Updated
PROBLEMS OUTPUT TERMINAL PORTS GITLENS COMMENTS DEBUG CONSOLE
powershell - finance-me-banking-finance
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-banking-finance> aws configure
AWS Access Key ID [**********MLWM]:
AWS Secret Access Key [**********AJUI]:
Default region name [ap-south-1]:
Default output format [json]:
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-banking-finance>
OUTLINE
TIMELINE
master
Launchpad
Live Share
Go Live

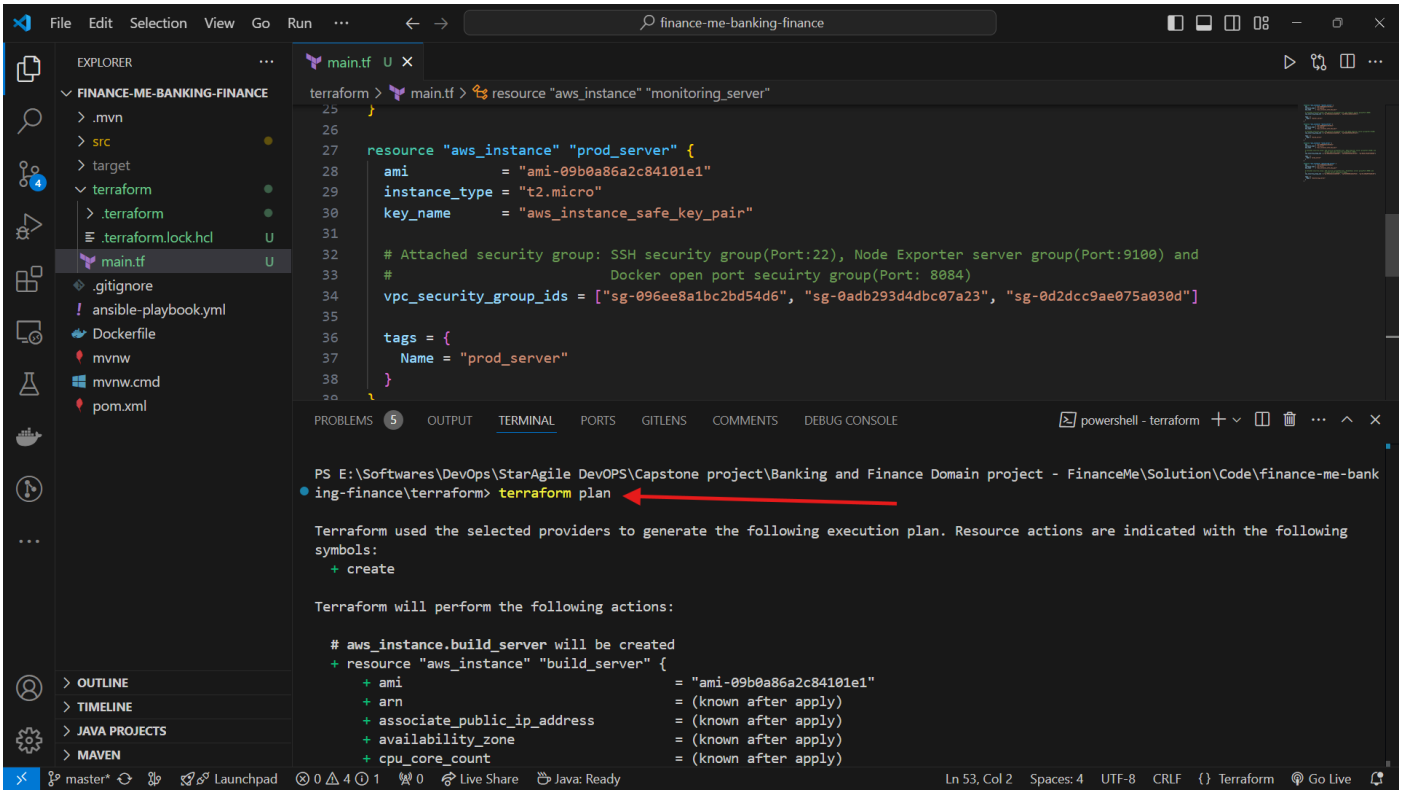

File Edit Selection View Go Run
Code
EXPLORER
CODE
finance-me-banking-finance
.mvn
src
.gitignore
ansible-playbook.yml
Dockerfile
mvnw
mvnw.cmd
pom.xml
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
Recent
Ideal Code from Claude E:\Softwares\DevOps\StarAgile DevOP...
Sir code E:\Softwares\DevOps\StarAgile DevOPS\Terraform\Proj...
Walkthroughs
Get Started with GitLens Updated
Get Started with Python Development Updated
Get Started with Jupyter Notebooks Updated
Get Started with WSL Updated
Get Started with C++ Development Updated
PROBLEMS OUTPUT TERMINAL PORTS GITLENS COMMENTS DEBUG CONSOLE
powershell - finance-me-banking-finance
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-banking-finance> terraform --version
Terraform v1.9.8
on windows_amd64
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-banking-finance>
OUTLINE
TIMELINE
master Launchpad 0 0 0 Live Share
Go Live

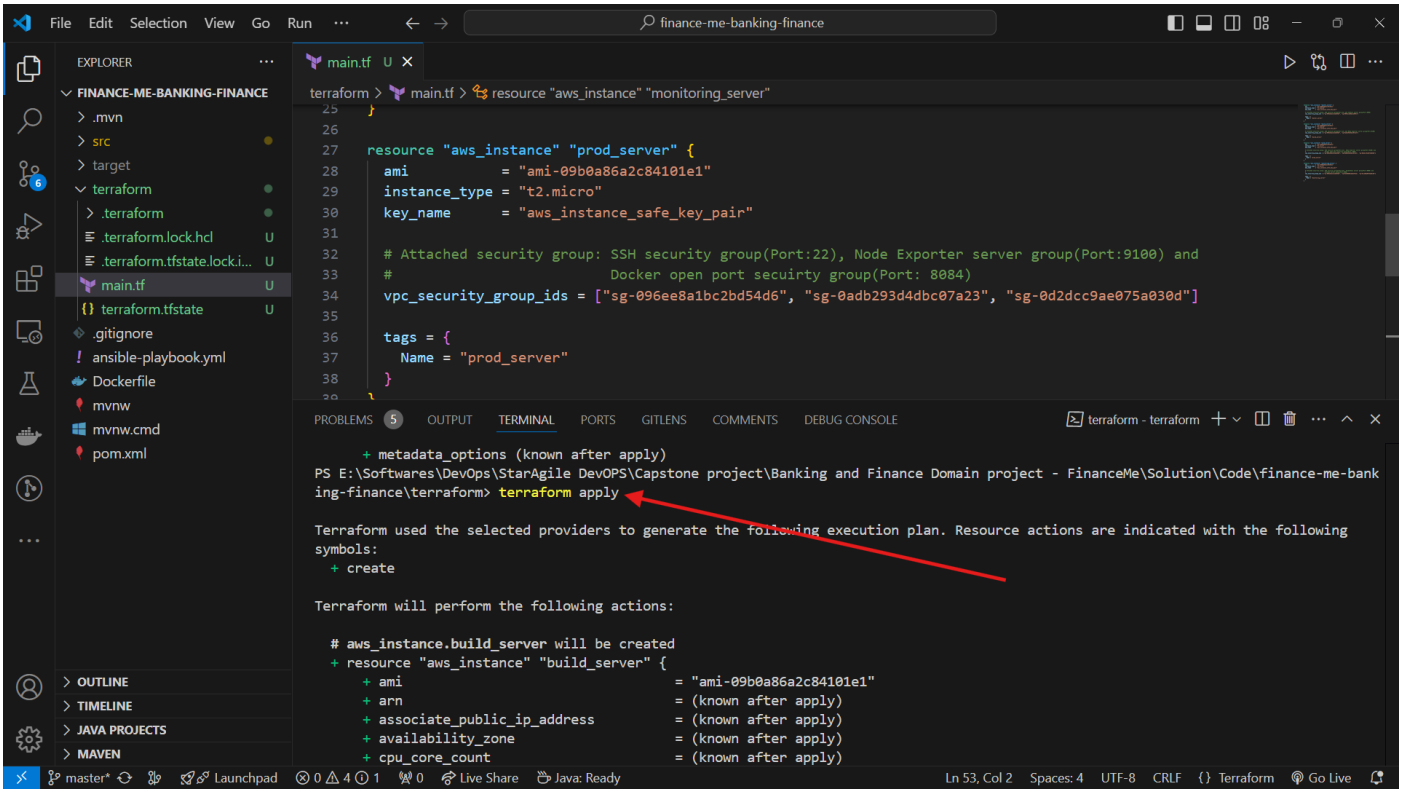

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#SecurityGroups:
aws Services Search [Alt+S]
Mumbai abhijitzende
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores New
Auto Scaling
Auto Scaling Groups
Settings
Security Groups (16) Info
Actions
Export security groups to CSV
Create security group
Find resources by attribute or tag
Name
Security group ID
Security group name
VPC ID
Description
sg-089835b2a91440b2e launch-wizard-1 vpc-067c947592b5dacbe launch-wizard
sg-0ba48ff68fd9e9eb0 MyKubernetesNodePortDefaultGroup vpc-067c947592b5dacbe Allowing traffi
sg-096ee8a1bc2bd54d6 - MySSHSeccurityGroup
Details
Inbound rules
Outbound rules
Sharing - new
VPC associations - new
Tags
Inbound rules (2)
Manage tags
Edit inbound rules
Search
Name
Security group rule...
IP version
Type
Protocol
Port range
sgr-0fa7d2d8ed8148d... IPv6 SSH TCP 22
sgr-04549e710ecd26d... IPv4 SSH TCP 22
CloudShell Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

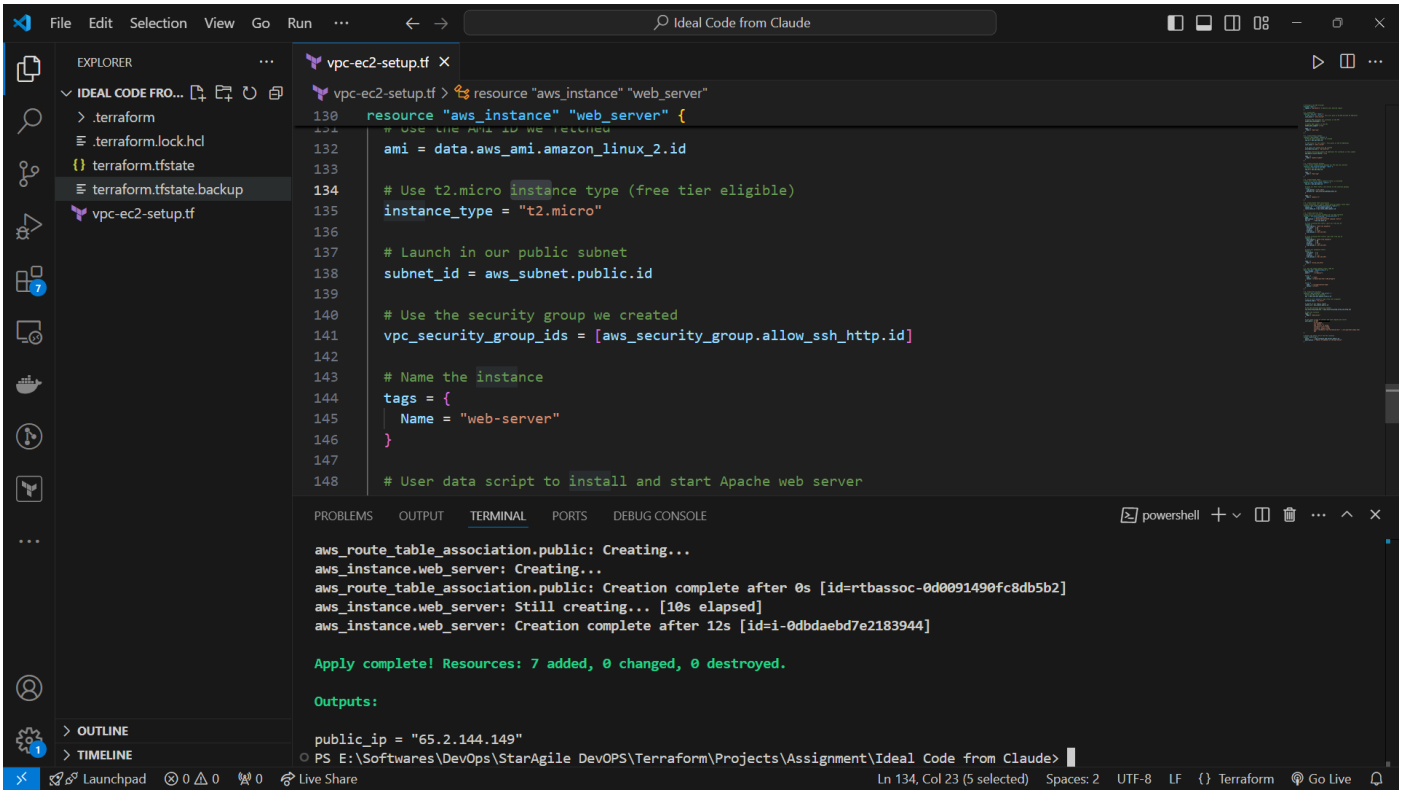

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:
Services
Search
[Alt+S]
Mumbai
abhijitzende
Search our full catalog including 1000s of application and OS images
Recents
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Li
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
Ubuntu Server 22.04 LTS (HVM), SSD Volume Type
Free tier eligible
ami-09b0a86a2c84101e1 (64-bit (x86)) / ami-0a87daabd88e93b1f (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs
Description
Ubuntu Server 22.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).
Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2024-09-27
Architecture
64-bit (x86)
AMI ID
ami-09b0a86a2c84101e1
Username
ubuntu
Verified provider
Summary
Number of instances
Info
1
Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...read more
ami-09b0a86a2c84101e1
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4
Cancel
Launch instance
Preview code
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

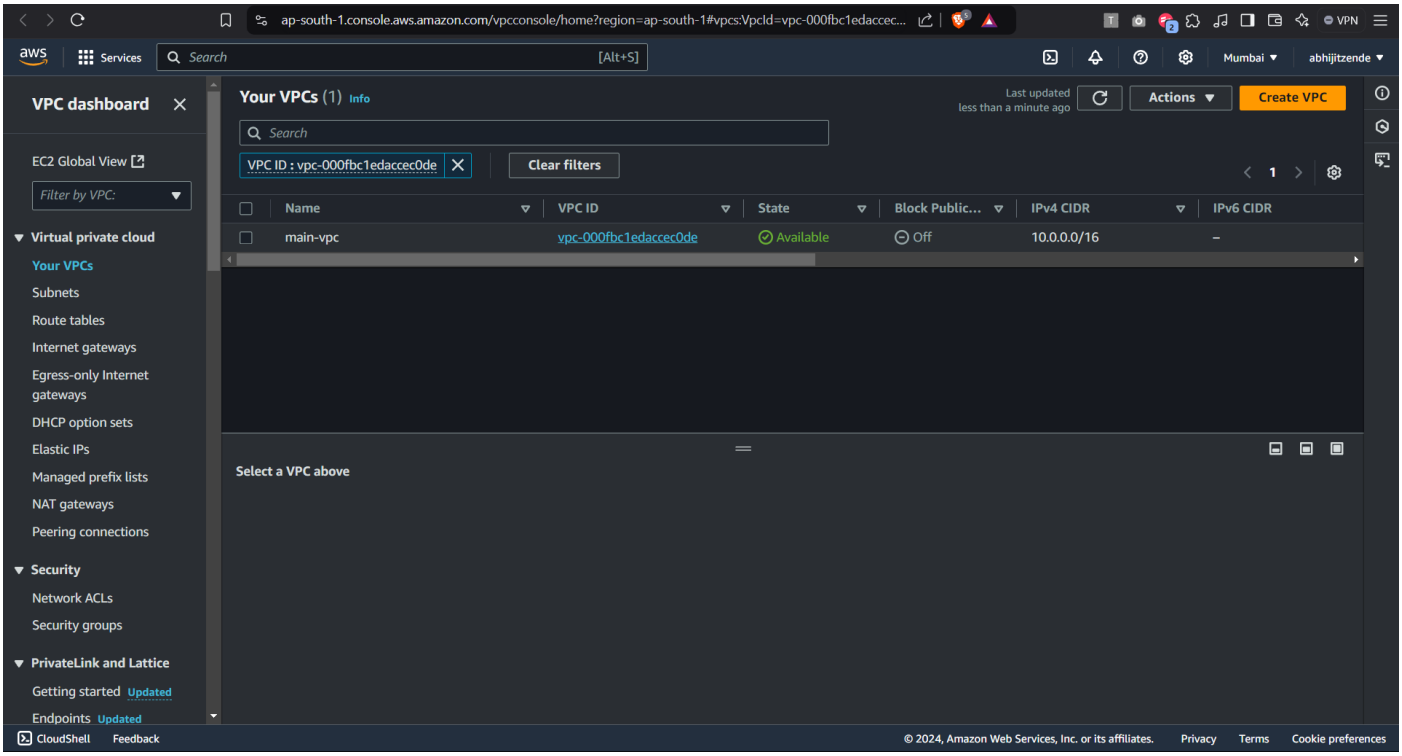

File Edit Selection View Go Run
finance-me-banking-finance
EXPLORER
FINANCE-ME-BANKING-FINANCE
.mvn
src
target
terraform
.terraform
.terraform.lock.hcl
main.tf
.gitignore
ansible-playbook.yml
Dockerfile
mvnw
mvnw.cmd
pom.xml
main.tf
terraform > main.tf > resource "aws_instance" "monitoring_server"
resource "aws_instance" "prod_server" {
ami = "ami-09b0a86a2c84101e1"
instance_type = "t2.micro"
key_name = "aws_instance_safe_key_pair"
# Attached security group: SSH security group(Port:22), Node Exporter server group(Port:9100) and
# Docker open port secuirty group(Port: 8084)
vpc_security_group_ids = ["sg-096ee8a1bc2bd54d6", "sg-0adb293d4dbc07a23", "sg-0d2dcc9ae075a030d"]
tags = {
Name = "prod_server"
}
PROBLEMS 5
OUTPUT
TERMINAL
PORTS
GITLENS
COMMENTS
DEBUG CONSOLE
powershell - terraform
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-banking-finance\terraform> terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.75.1...
- Installed hashicorp/aws v5.75.1 (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.
OUTLINE
TIMELINE
JAVA PROJECTS
MAVEN
master*
Launchpad
Live Share
Java: Ready
Ln 53, Col 2
Spaces: 4
UTF-8
CRLF
Terraform
Go Live

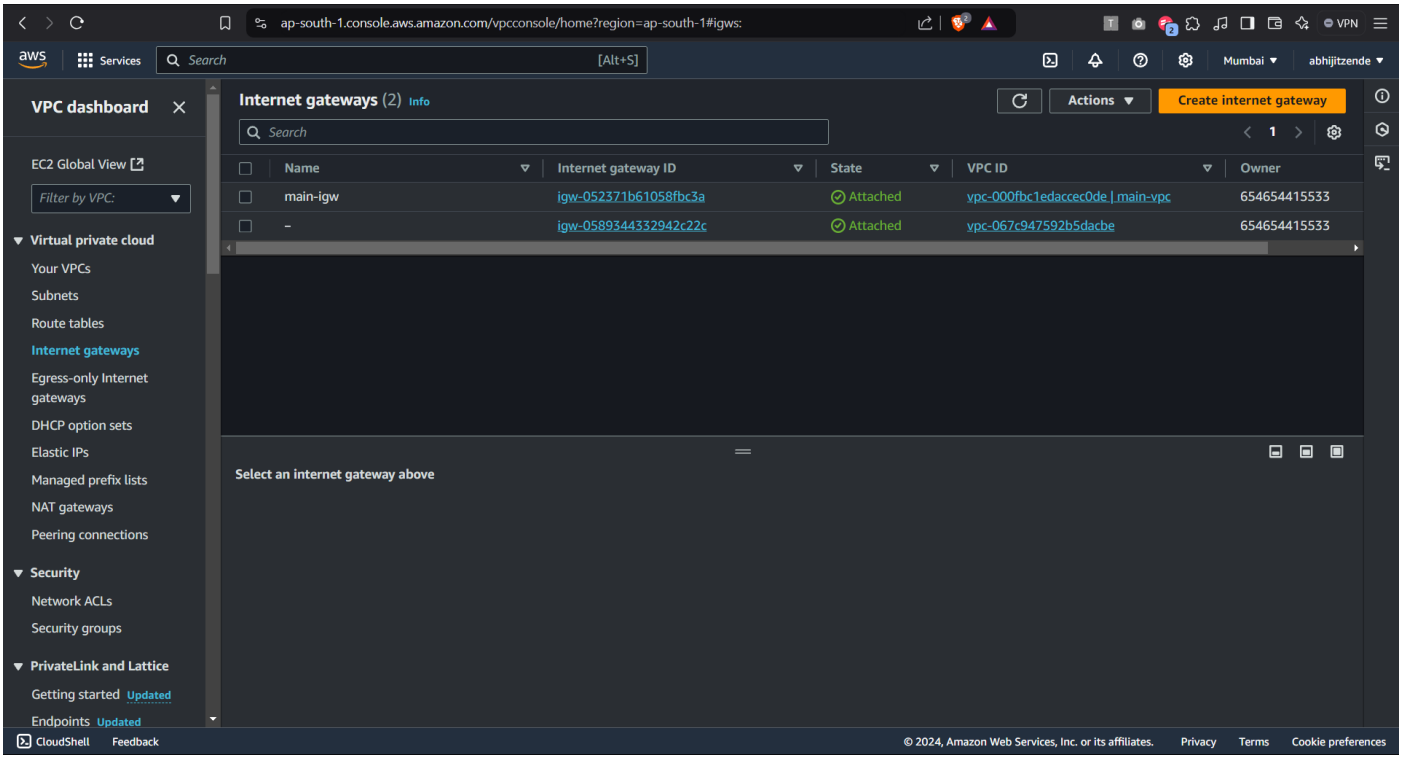

File Edit Selection View Go Run
finance-me-banking-finance
EXPLORER
FINANCE-ME-BANKING-FINANCE
.mvn
src
target
terraform
.terraform
.terraform.lock.hcl
main.tf
.gitignore
ansible-playbook.yml
Dockerfile
mvnw
mvnw.cmd
pom.xml
main.tf U
terraform > main.tf > resource "aws_instance" "monitoring_server"
resource "aws_instance" "prod_server" {
ami = "ami-09b0a86a2c84101e1"
instance_type = "t2.micro"
key_name = "aws_instance_safe_key_pair"
# Attached security group: SSH security group(Port:22), Node Exporter server group(Port:9100) and
# Docker open port secuirty group(Port: 8084)
vpc_security_group_ids = ["sg-096ee8a1bc2bd54d6", "sg-0adb293d4dbc07a23", "sg-0d2dcc9ae075a030d"]
tags = {
Name = "prod_server"
}
PROBLEMS 5 OUTPUT TERMINAL PORTS GITLENS COMMENTS DEBUG CONSOLE
powershell - terraform
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.
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-bank
ing-finance\terraform> terraform fmt
main.tf
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-bank
ing-finance\terraform> terraform validate
Success! The configuration is valid.
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-bank
ing-finance\terraform>
OUTLINE
TIMELINE
JAVA PROJECTS
MAVEN
master* Launchpad 0 4 1 0 Live Share Java: Ready
Ln 53, Col 2 Spaces: 4 UTF-8 CRLF Terraform Go Live

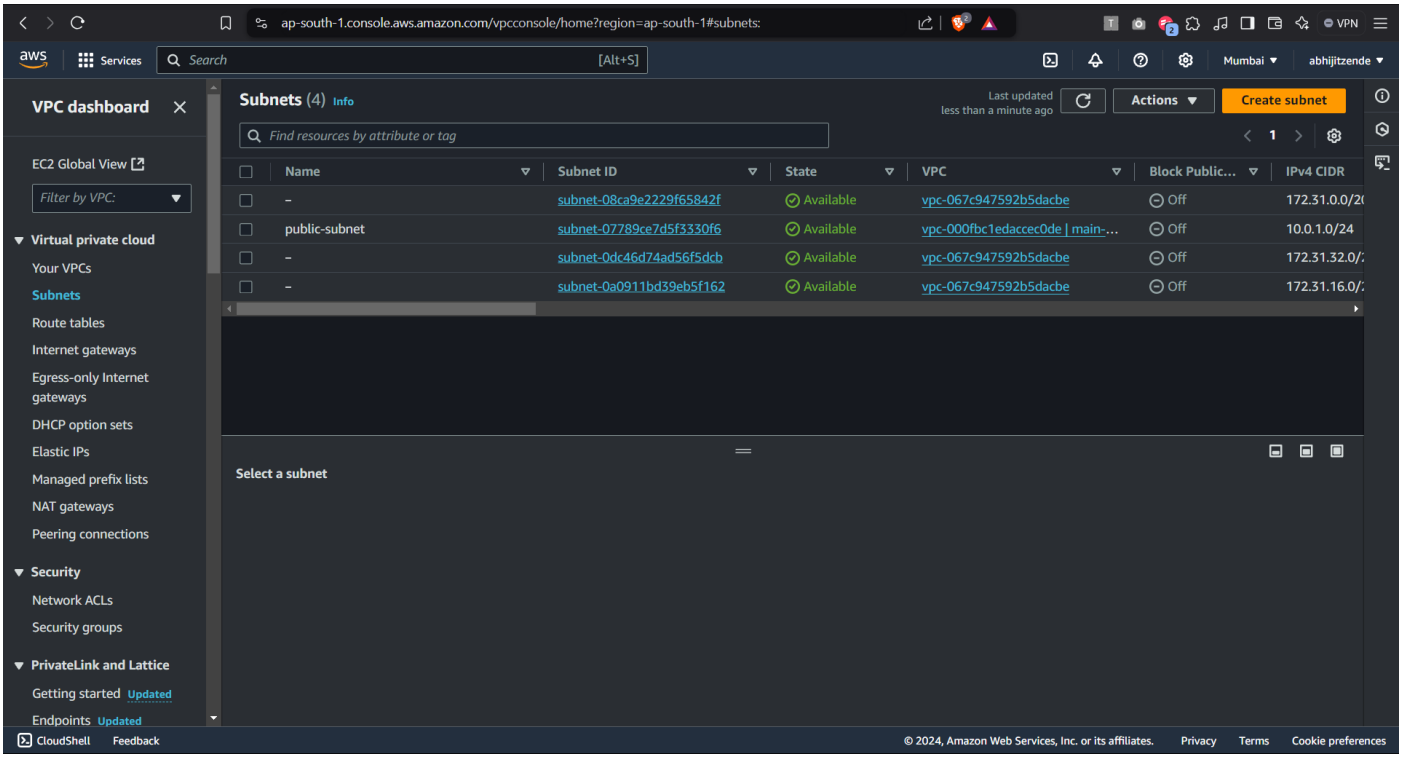

File Edit Selection View Go Run
finance-me-banking-finance
EXPLORER
FINANCE-ME-BANKING-FINANCE
.mvn
src
target
terraform
.terraform
.terraform.lock.hcl
main.tf
.gitignore
ansible-playbook.yml
Dockerfile
mvnw
mvnw.cmd
pom.xml
main.tf U
terraform > main.tf > resource "aws_instance" "monitoring_server"
resource "aws_instance" "prod_server" {
ami = "ami-09b0a86a2c84101e1"
instance_type = "t2.micro"
key_name = "aws_instance_safe_key_pair"
# Attached security group: SSH security group(Port:22), Node Exporter server group(Port:9100) and
# Docker open port secuirty group(Port: 8084)
vpc_security_group_ids = ["sg-096ee8a1bc2bd54d6", "sg-0adb293d4dbc07a23", "sg-0d2dcc9ae075a030d"]
tags = {
Name = "prod_server"
}
PROBLEMS 5 OUTPUT TERMINAL PORTS GITLENS COMMENTS DEBUG CONSOLE
powershell - terraform
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-bank
ing-finance\terraform> terraform plan
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_instance.build_server will be created
+ resource "aws_instance" "build_server" {
+ ami = "ami-09b0a86a2c84101e1"
+ arn = (known after apply)
+ associate_public_ip_address = (known after apply)
+ availability_zone = (known after apply)
+ cpu_core_count = (known after apply)
OUTLINE
TIMELINE
JAVA PROJECTS
MAVEN
master* Launchpad 0 4 1 0 Live Share Java: Ready
Ln 53, Col 2 Spaces: 4 UTF-8 CRLF Terraform Go Live

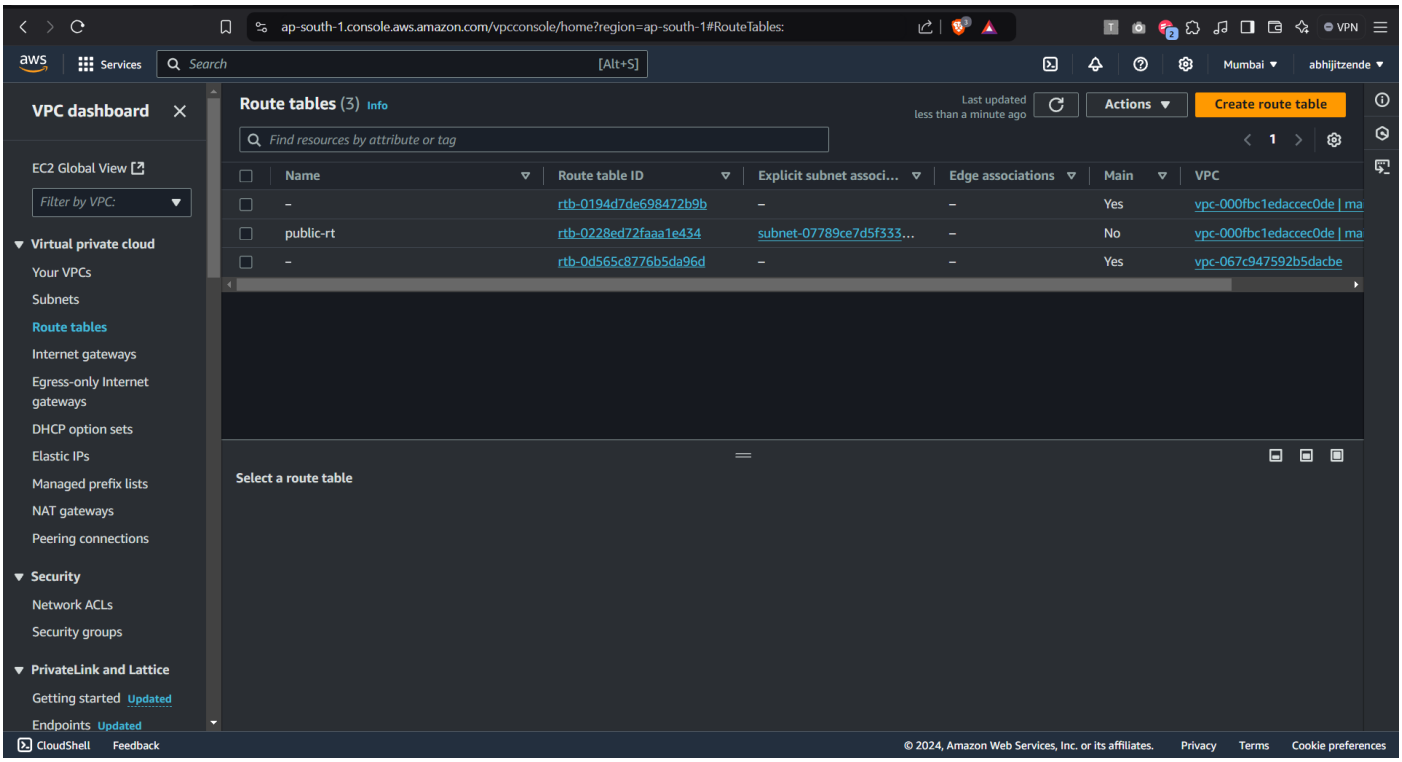

File Edit Selection View Go Run
finance-me-banking-finance
EXPLORER
FINANCE-ME-BANKING-FINANCE
.mvn
src
target
terraform
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.i...
main.tf
terraform.tfstate
.gitignore
ansible-playbook.yml
Dockerfile
mvnw
mvnw.cmd
pom.xml
main.tf U
terraform > main.tf > resource "aws_instance" "monitoring_server"
resource "aws_instance" "prod_server" {
ami = "ami-09b0a86a2c84101e1"
instance_type = "t2.micro"
key_name = "aws_instance_safe_key_pair"
# Attached security group: SSH security group(Port:22), Node Exporter server group(Port:9100) and
# Docker open port secuirty group(Port: 8084)
vpc_security_group_ids = ["sg-096ee8a1bc2bd54d6", "sg-0adb293d4dbc07a23", "sg-0d2dcc9ae075a030d"]
tags = {
Name = "prod_server"
}
PROBLEMS 5 OUTPUT TERMINAL PORTS GITLENS COMMENTS DEBUG CONSOLE
terraform - terraform
+ metadata_options (known after apply)
PS E:\Softwares\DevOps\StarAgile DevOPS\Capstone project\Banking and Finance Domain project - FinanceMe\Solution\Code\finance-me-banking-finance\terraform> terraform apply
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_instance.build_server will be created
+ resource "aws_instance" "build_server" {
+ ami = "ami-09b0a86a2c84101e1"
+ arn = (known after apply)
+ associate_public_ip_address = (known after apply)
+ availability_zone = (known after apply)
+ cpu_core_count = (known after apply)
OUTLINE
TIMELINE
JAVA PROJECTS
MAVEN
master* Launchpad 0 4 1 0 Live Share Java: Ready
Ln 53, Col 2 Spaces: 4 UTF-8 CRLF Terraform Go Live

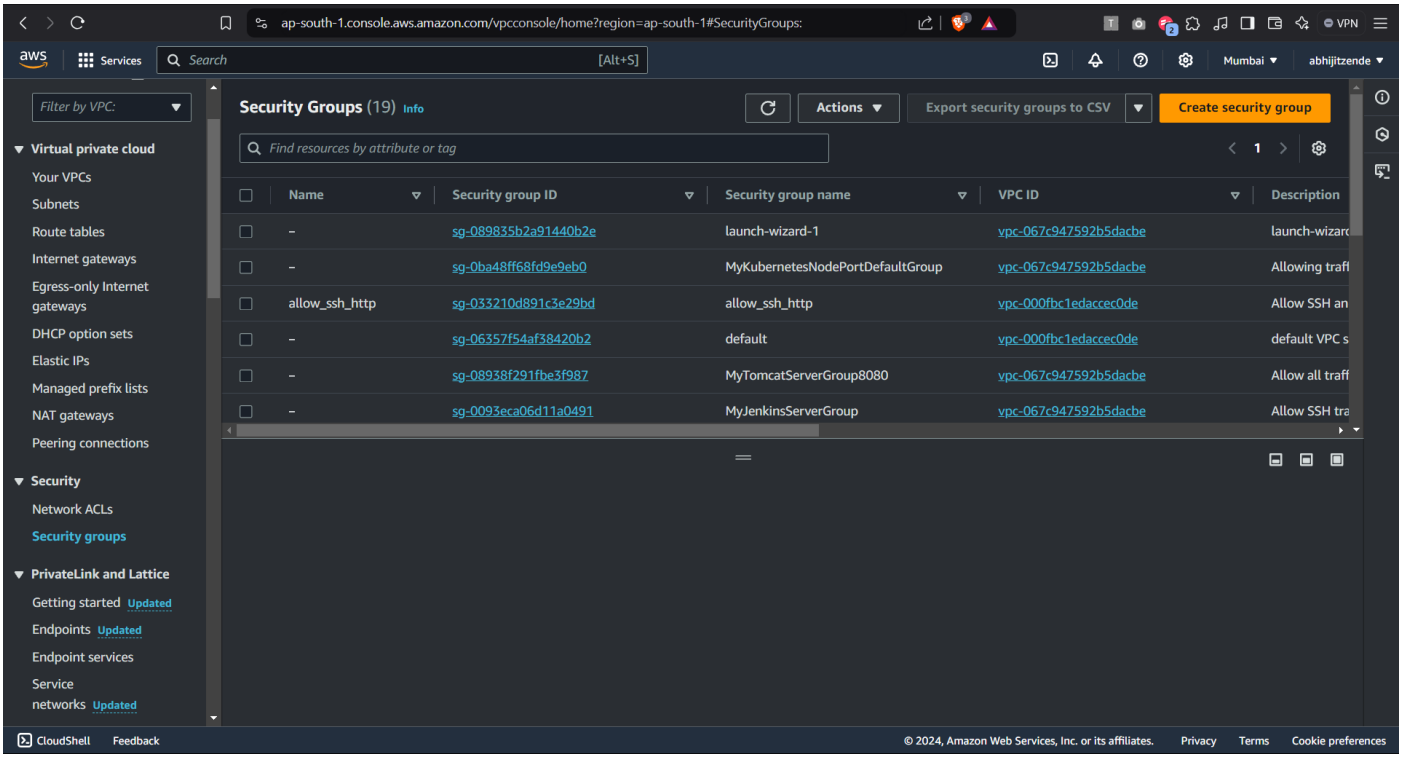

File Edit Selection View Go Run
Ideal Code from Claude
EXPLORER
IDEAL CODE FRO...
.terraform
.terraform.lock.hcl
terraform.tfstate
terraform.tfstate.backup
vpc-ec2-setup.tf
vpc-ec2-setup.tf
vpc-ec2-setup.tf > resource "aws_instance" "web_server"
resource "aws_instance" "web_server" {
ami = data.aws_ami.amazon_linux_2.id
# Use t2.micro instance type (free tier eligible)
instance_type = "t2.micro"
# Launch in our public subnet
subnet_id = aws_subnet.public.id
# Use the security group we created
vpc_security_group_ids = [aws_security_group.allow_ssh_http.id]
# Name the instance
tags = {
Name = "web-server"
}
# User data script to install and start Apache web server
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE
powershell
aws_route_table_association.public: Creating...
aws_instance.web_server: Creating...
aws_route_table_association.public: Creation complete after 0s [id=rtbassoc-0d0091490fc8db5b2]
aws_instance.web_server: Still creating... [10s elapsed]
aws_instance.web_server: Creation complete after 12s [id=i-0dbdaebd7e2183944]
Apply complete! Resources: 7 added, 0 changed, 0 destroyed.
Outputs:
public_ip = "65.2.144.149"
PS E:\Softwares\DevOps\StarAgile DevOPS\Terraform\Projects\Assignment\Ideal Code from Claude>
OUTLINE
TIMELINE
Launchpad 0 0 0 Live Share
Ln 134, Col 23 (5 selected) Spaces: 2 UTF-8 LF Terraform Go Live

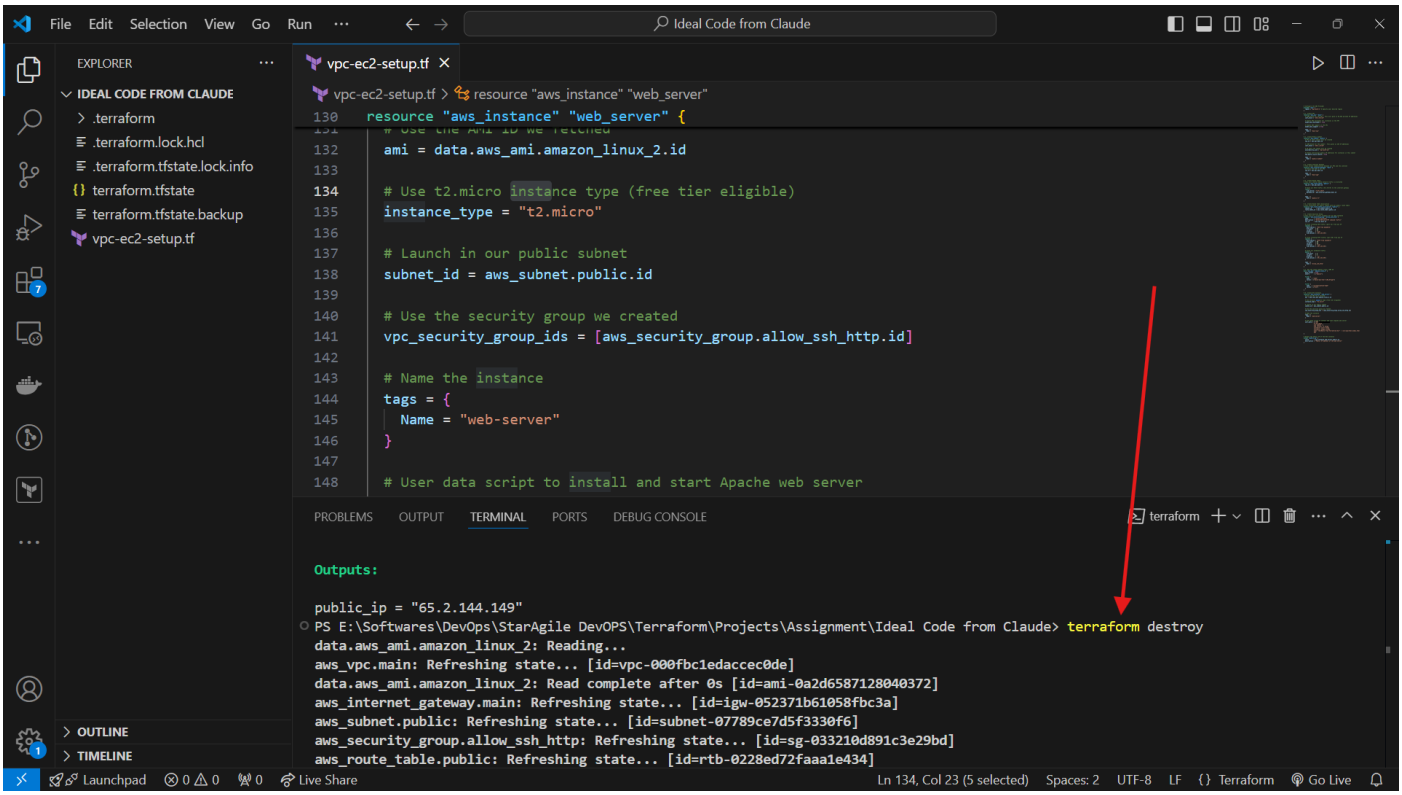

ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#vpcs:VpcId=vpc-000fbc1edaccec...
aws
Services
Search
[Alt+S]
Mumbai
abhijitzende
VPC dashboard
EC2 Global View
Filter by VPC:
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only Internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Your VPCs (1) Info
Last updated less than a minute ago
Actions
Create VPC
Search
VPC ID : vpc-000fbc1edaccec0de
Clear filters
Name
VPC ID
State
Block Public...
IPv4 CIDR
IPv6 CIDR
main-vpc
vpc-000fbc1edaccec0de
Available
Off
10.0.0.0/16
–
Select a VPC above
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#igws:
aws
Services
Search
[Alt+S]
Mumbai
abhijitzende
VPC dashboard
EC2 Global View
Filter by VPC:
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only Internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Internet gateways (2) Info
Actions
Create internet gateway
Search
Name
Internet gateway ID
State
VPC ID
Owner
main-igw
igw-052371b61058fbc3a
Attached
vpc-000fbc1edaccec0de | main-vpc
654654415533
–
igw-0589344332942c22c
Attached
vpc-067c947592b5dacbe
654654415533
Select an internet gateway above
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#subnets:
aws
Services
Search
[Alt+S]
Mumbai
abhijitzende
VPC dashboard
EC2 Global View
Filter by VPC:
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only Internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Subnets (4) Info
Last updated less than a minute ago
Actions
Create subnet
Find resources by attribute or tag
Name
Subnet ID
State
VPC
Block Public...
IPv4 CIDR
- subnet-08ca9e2229f65842f Available vpc-067c947592b5dacbe Off 172.31.0.0/20
public-subnet subnet-07789ce7d5f3330f6 Available vpc-000fbc1edaccec0de | main-... Off 10.0.1.0/24
- subnet-0dc46d74ad56f5dcb Available vpc-067c947592b5dacbe Off 172.31.32.0/
- subnet-0a0911bd39eb5f162 Available vpc-067c947592b5dacbe Off 172.31.16.0/
Select a subnet
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#RouteTables:
aws
Services
Search
[Alt+S]
Mumbai
abhijitzende
VPC dashboard
EC2 Global View
Filter by VPC:
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only Internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Route tables (3) Info
Last updated less than a minute ago
Actions
Create route table
Find resources by attribute or tag
Name
Route table ID
Explicit subnet associ...
Edge associations
Main
VPC
- rtb-0194d7de698472b9b - - Yes vpc-000fbc1edaccec0de | ma
public-rt rtb-0228ed72faaa1e434 subnet-07789ce7d5f333... - No vpc-000fbc1edaccec0de | ma
- rtb-0d565c8776b5da96d - - Yes vpc-067c947592b5dacbe
Select a route table
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#SecurityGroups:
Services
Search
[Alt+S]
Mumbai
abhijitzende
Filter by VPC:
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only Internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Endpoint services
Service networks Updated
Security Groups (19) Info
Actions
Export security groups to CSV
Create security group
Find resources by attribute or tag
Name
Security group ID
Security group name
VPC ID
Description
- sg-089835b2a91440b2e launch-wizard-1 vpc-067c947592b5dacbe launch-wizard
- sg-0ba48ff68fd9e9eb0 MyKubernetesNodePortDefaultGroup vpc-067c947592b5dacbe Allowing traff
allow_ssh_http sg-033210d891c3e29bd allow_ssh_http vpc-000fbc1edaccec0de Allow SSH an
- sg-06357f54af38420b2 default vpc-000fbc1edaccec0de default VPC s
- sg-08938f291fbe3f987 MyTomcatServerGroup8080 vpc-067c947592b5dacbe Allow all traff
- sg-0093eca06d11a0491 MyJenkinsServerGroup vpc-067c947592b5dacbe Allow SSH tra
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


File Edit Selection View Go Run
Ideal Code from Claude
EXPLORER
IDEAL CODE FROM CLAUDE
.terraform
.terraform.lock.hcl
.terraform.tfstate.lock.info
terraform.tfstate
terraform.tfstate.backup
vpc-ec2-setup.tf
vpc-ec2-setup.tf
vpc-ec2-setup.tf > resource "aws_instance" "web_server"
130 resource "aws_instance" "web_server" {
132 ami = data.aws_ami.amazon_linux_2.id
134 # Use t2.micro instance type (free tier eligible)
135 instance_type = "t2.micro"
137 # Launch in our public subnet
138 subnet_id = aws_subnet.public.id
140 # Use the security group we created
141 vpc_security_group_ids = [aws_security_group.allow_ssh_http.id]
143 # Name the instance
144 tags = {
145 Name = "web-server"
146 }
148 # User data script to install and start Apache web server
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE
terraform
Outputs:
public_ip = "65.2.144.149"
PS E:\Softwares\DevOps\StarAgile DevOPS\Terraform\Projects\Assignment\Ideal Code from Claude> terraform destroy
data.aws_ami.amazon_linux_2: Reading...
aws_vpc.main: Refreshing state... [id=vpc-000fbc1edaccec0de]
data.aws_ami.amazon_linux_2: Read complete after 0s [id=ami-0a2d6587128040372]
aws_internet_gateway.main: Refreshing state... [id=igw-052371b61058fbc3a]
aws_subnet.public: Refreshing state... [id=subnet-07789ce7d5f3330f6]
aws_security_group.allow_ssh_http: Refreshing state... [id=sg-033210d891c3e29bd]
aws_route_table.public: Refreshing state... [id=rtb-0228ed72faaa1e434]
OUTLINE
TIMELINE
Launchpad
Live Share
Ln 134, Col 23 (5 selected)
Spaces: 2
UTF-8
LF
Terraform
Go Live