Terraform Modules

VPC Module

Let's start with the VPC module.

1. Create *vpc* folder.
2. Inside *vpc* folder create *main.tf* file.
3. Create resources for the following:
    - aws_vpc
    - for public
        - aws_subnet
        - aws_internet_gateway
        - aws_route_table
        - aws_route
        - aws_route_table_association
    - for private
        - aws_subnet
        - aws_route_table
        - aws_route_table_association
4. Click code for reference.
5. Now the *main.tf* file definition for VPC has been created.
6. Now we will create *variables.tf* file inside *vpc* folder for declaring variables.
7. Declare the following variables:
    - vpc-properties
8. Click code for reference.
9. We have completed declaring variables for the VPC module.
10. Now we will declare outputs for the VPC module.
11. Create *outputs.tf* file inside the same *vpc* folder.
12. Add the following output:
    - vpc-id
    - vpc-private-subnets
    - vpc-public-subnets
13. Click code for reference.
14. Now we have completed defining the **VPC Module**.

S3 Module

We will use S3 to store and access .env files for ecs containers.

1. Create *s3* folder.
2. Inside *s3* folder, create *main.tf* file.
3. Define the following resources:
    - aws_s3_bucket
    - aws_s3_bucket_versioning

4. Click code for reference.
5. We have completed defining *main.tf* file.
6. Now create *variables.tf* file.
7. Define the following variables:
    - s3-properties
8. Click code for reference:
9. *variables.tf* file has been declared.
10. Now create *output.tf* file.
11. Inside *output.tf* file, define the following outputs:
    - s3-bucket-id
12. Click code for reference.
13. We have completed defining the **S3 Module**.

RDS Module

For database storage, we will use MySQL RDS.

1. Create *rds* folder.
2. Inside *rds* folder, create *data.tf* file.
3. Define the following data:
    - aws_ami
4. Click code for reference.
5. We have completed defining the *data.tf* file.
6. Now we will create *main.tf* file.
7. Define the following resources:
    - for database
        - aws_db_subnet_group
        - aws_security_group
        - aws_db_instance
    - for bastion
        - aws_security_group
        - aws_key_pair
8. Click code for reference.
9. The definition of *main.tf* file is completed.
10. Now we will create *variables.tf* file.
11. Inside *variables.tf* define the following variables:
    - vpc-id
    - vpc-public-subnets
    - vpc-private-subnets
    - database-properties
    - bastion-properties
12. Click code for reference.
13. Variables have been declared, now we will define the output.
14. Create *outputs.tf* file.
15. Inside the *outputs.tf* file, define the following output:
    - DB_HOST

    - bastion-host-ip
16. Click code for reference.
17. We have completed defining the **RDS Module**.

ECR Module

Let's start with the ECR Module

1. Create *ecr* folder.
2. Inside *ecr* folder, create *main.tf* file.
3. In *main.tf* file, define the following resources:
    - aws_ecr_repository
4. Click code for reference.
5. The definition of *main.tf* file is complete.
6. Now we will create *variables.tf* file and declare the following variables:
    - ecr-repository-name
7. Click code for reference.
8. The declaration of variables is completed.
9. Now we will create *outputs.tf* file and define the following output:
    - repository-url
10. Click code for reference.
11. The definition of *outputs.tf* file is complete.
12. We have completed creating the **ECR Module**.

Load Balancer Module

Let's start with the Load Balancer Module

1. Create *load-balancer* folder.
2. Inside *load-balancer* folder, create *main.tf* file.
3. In the *main.tf* file, define the following resources:
    - aws_security_group
    - aws_lb
    - aws_lb_target_group
    - aws_lb_listener
4. Click code for reference.
5. The *main.tf* file for Load Balancer Module has been defined.
6. Now we will define variables.tf file.
7. Create *variables.tf* file inside *load-balancer* folder.
8. Define the following variables:
    - vpc-id
    - vpc-public-subnets
    - vpc-private-subnets
    - load-balancer-properties
9. Click code for reference.

10. The *variable.tf* file for Load Balancer Module has been defined.
11. Now we will define outputs.tf file.
12. Create *outputs.tf* file inside *load-balancer* folder.
13. Define the following variables:
    - load-balancer-sg-id
    - load-balancer-tg-arn
14. Click code for reference.
15. We have completed defining the **Load Balancer Module**.

ECS Module

Let's start with the ECS Module

1. Create *ecs* folder.
2. Inside *ecs* folder, create *data.tf* file.
3. In *data.tf* file, define the following data:
    - aws_iam_policy_document
4. Click code for reference.
5. Definition of *data.tf* file is completed.
6. Now we will define *main.tf* file.
7. Inside *ecs* folder create *main.tf* file.
8. In the *main.tf* file, define the following resources:
    - for cluster
        - aws_ecs_cluster
    - for task
        - aws_iam_role
        - aws_iam_role_policy_attachment
        - aws_ecs_task_definition
    - for service
        - aws_security_group
        - aws_ecs_service
9. Click code for reference.
10. The *main.tf* file for ECS has been defined.
11. Now we will define variables.tf file.
12. Create *variables.tf* file inside *ecs* folder.
13. Define the following variables:
    - availability-zones
    - vpc-id
    - vpc-public-subnets
    - vpc-private-subnets
    - ecr-repository-url
    - repository-url
    - ecs-properties
    - ecs-container-definition
    - load-balancer-tg-arn
    - load-balancer-sg-id

14. Click code for reference.
15. We have completed defining the **ECS Module**.

EKS Module

Let's start with the EKS Module

1. Create *eks* folder.
2. Inside *eks* folder create *data.tf* file.
3. In *data.tf* file, define the following data:
    - aws_iam_policy_document
4. Click code for reference.
5. Definition of *data.tf* file is completed.
6. Now we will define *main.tf* file.
7. Inside *eks* folder create *main.tf* file.
8. In the *main.tf* file, define the following resources:
    - for cluster
        - aws_iam_role
        - aws_iam_role_policy_attachment
        - aws_eks_cluster
    - for node
        - aws_iam_role
        - aws_iam_role_policy_attachment
        - aws_iam_role_policy_attachment
        - aws_iam_role_policy_attachment
        - aws_eks_node_group
9. Click code for reference.
10. The *main.tf* file for EKS has been defined.
11. Now we will define variables.tf file.
12. Create *variables.tf* file inside *eks* folder.
13. Define the following variables:
    - vpc-public-subnets
    - vpc-private-subnets
    - eks-properties
14. Click code for reference.
15. We have completed defining the **EKS Module**.
