**ALL Learnings Doc**

**Explain the following block of Terraform code?**

resource "aws\_instance" "gfg-ec2" {

ami = "ami-0c94855ba95c286c99"

instance\_type = "t2.micro"

}

Let's break down the Terraform HCL (HashiCorp Configuration Language) syntax used in the provided code:

* **resource**: This keyword indicates that you're declaring a new resource. In Terraform, resources represent infrastructure components such as instances, databases, networks, etc.
* **"aws\_instance"**: This is the resource type. In this case, it's an AWS EC2 instance. Terraform supports a wide range of providers, and "aws\_instance" indicates that we're creating a resource using the AWS provider.
* **"gfg-ec2"**: This is the name of the resource instance. It's a user-defined name for this specific instance of the "aws\_instance" resource. It must be unique within your Terraform configuration.
* **{}**: This block contains the configuration for the resource. All configuration settings specific to this resource instance are enclosed within these braces.
* **ami**: This is one of the configuration parameters for the EC2 instance. It specifies the Amazon Machine Image (AMI) ID that will be used to launch this instance. AMIs are pre-configured templates for virtual machines.
* **instance\_type**: This is another configuration parameter for the EC2 instance. It specifies the type of the instance to be launched. In this case, it's "t2.micro", which is a type of small, general-purpose instance available in AWS.
* **"ami-0c94855ba95c286c99" and "t2.micro"**: These are the values assigned to the ami and instance\_type parameters, respectively. They are literal string values representing the AMI ID and instance type.

Overall, this Terraform configuration defines an AWS EC2 instance resource named "gfg-ec2" using the specified AMI and instance type. When you apply this configuration using Terraform, it will create an EC2 instance in your AWS account with the specified characteristics.

**List of Configuration parameters for AWS EC2 Instance resource block**

When configuring an AWS EC2 instance in a Terraform **resource** block, there are several parameters you can use to customize its behavior. Here's a list of some common configurations you might include:

1. **ami**: The ID of the Amazon Machine Image (AMI) to use for the instance.
2. **instance\_type**: The type of EC2 instance to launch, such as **"t2.micro"**, **"m5.large"**, etc.
3. **subnet\_id**: The ID of the subnet to launch the instance into.
4. **vpc\_security\_group\_ids**: A list of security group IDs to associate with the instance.
5. **key\_name**: The name of the key pair to use for SSH access to the instance.
6. **user\_data**: Script or data to be executed on the instance at launch.
7. **tags**: A map of tags to apply to the instance.
8. **associate\_public\_ip\_address**: Whether the instance should have a public IP address associated with it.
9. **ebs\_optimized**: Whether the instance should have optimized EBS (Elastic Block Store) performance.
10. **root\_block\_device**: Configuration for the root EBS volume, including size, type, and delete on termination behavior.
11. **ebs\_block\_device**: Additional EBS volumes to attach to the instance.
12. **iam\_instance\_profile**: The IAM instance profile to associate with the instance.
13. **disable\_api\_termination**: Whether the instance can be terminated using the AWS Management Console, APIs, etc.
14. **instance\_initiated\_shutdown\_behavior**: The behavior when the instance is shut down (e.g., **"stop"**, **"terminate"**).

These are just some of the common configurations available for an AWS EC2 instance in Terraform. Depending on your specific requirements, there may be additional parameters you can use to customize the instance further. You can refer to the Terraform documentation for the AWS provider for a comprehensive list of available configurations and their descriptions.