**Manage Multiple Environments using Terragrunt**

The objective of managing multiple environments with Terragrunt is to streamline the management of infrastructure across different environments (such as development, staging, and production).

Requirements:

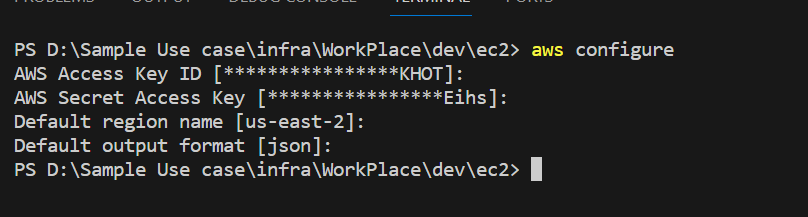
1. AWS cli
2. terragrunt

Solution:

**Configure AWS cli:**

Install AWS cli and configure IAM credentials using this below command.

**aws configure**

****

**Install Terragrunt:**

Install Terragrunt in windows using choco using this below command.

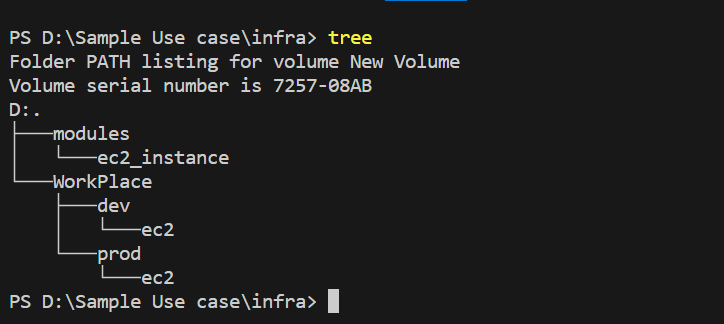
**choco install terragrunt**

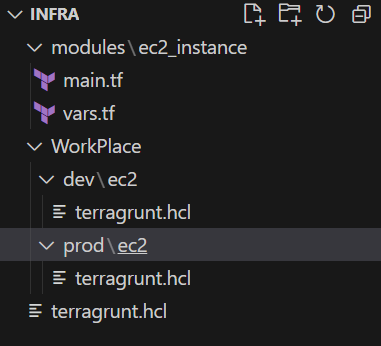
****

**Terragrunt:**

Terragrunt is a thin wrapper for Terraform that provides extra tools for keeping your Terraform configurations DRY (Don't Repeat Yourself). With Terragrunt, you can easily manage remote states and multiple environments. It also helps you keep your codebase clean and organized.

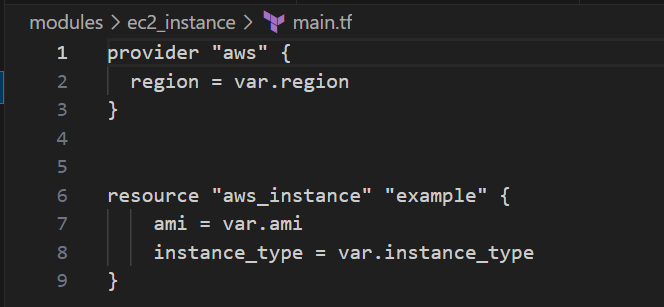
**File Structure:**

****

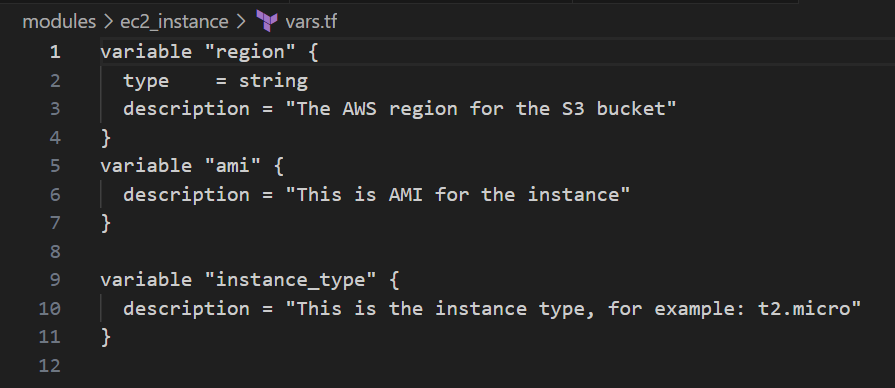
****

* For this we need to create a module directory that contains our resource configurations.
* Next we need to create a separate directory for different environments like dev and prod.
* That environment directories only contains terragrunt.hcl file.

**Module/ec2\_instane/main.tf**

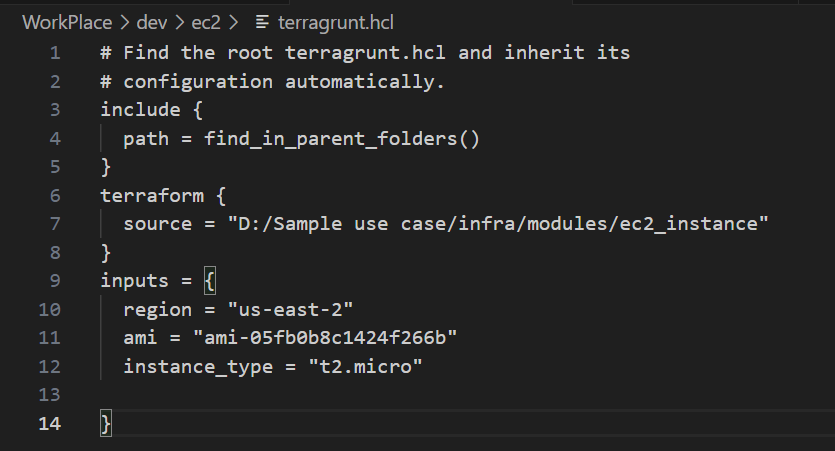
****

**Module/ec2\_instane/vars.tf**

****

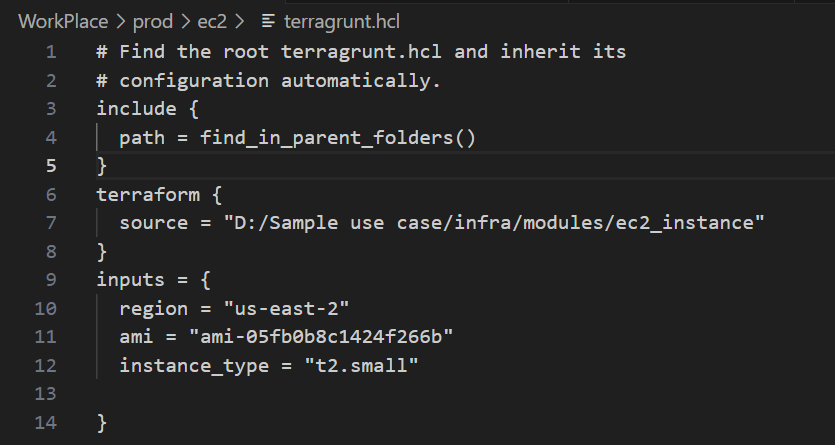
* This code for creating a EC2 using terragrunt as module.

**WorkPlace/dev/ec2/terragrunt.hcl**

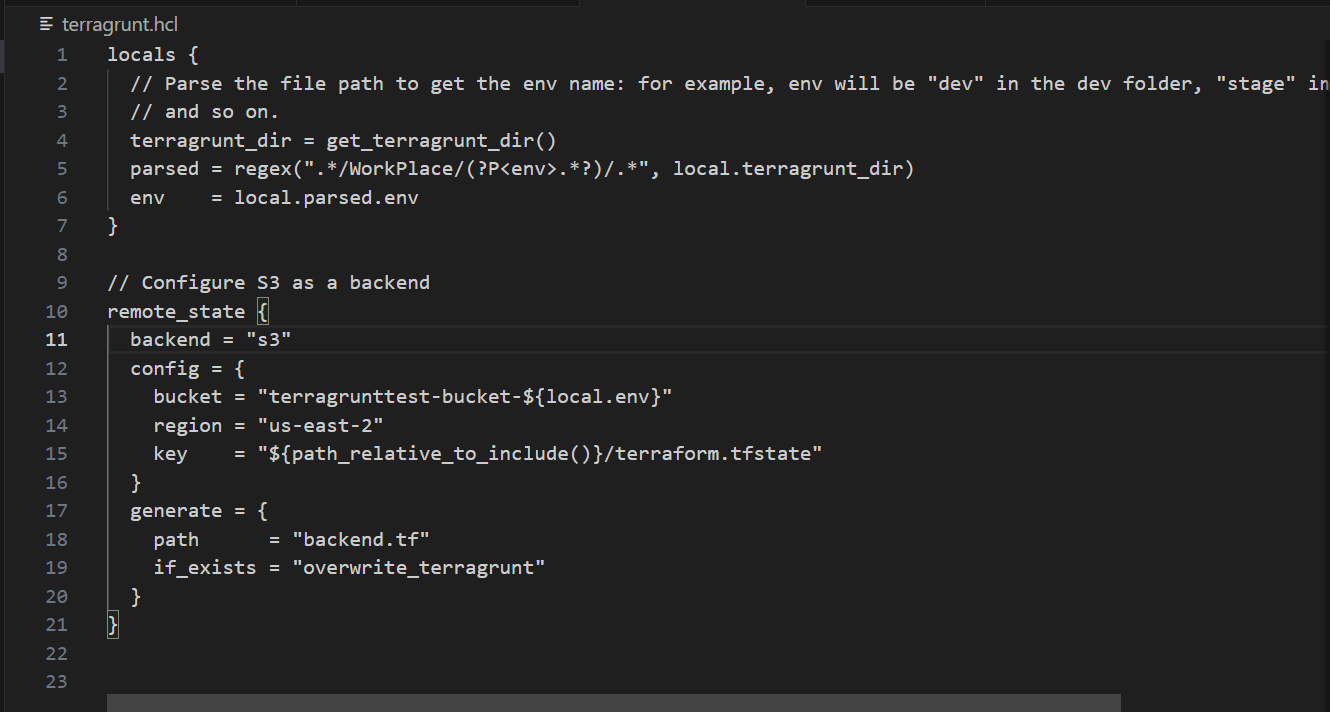
****

This code .hcl is used to create ec2 by inherit the module ec2 files.

**WorkPlace/dev/ec2/terragrunt.hcl**

****

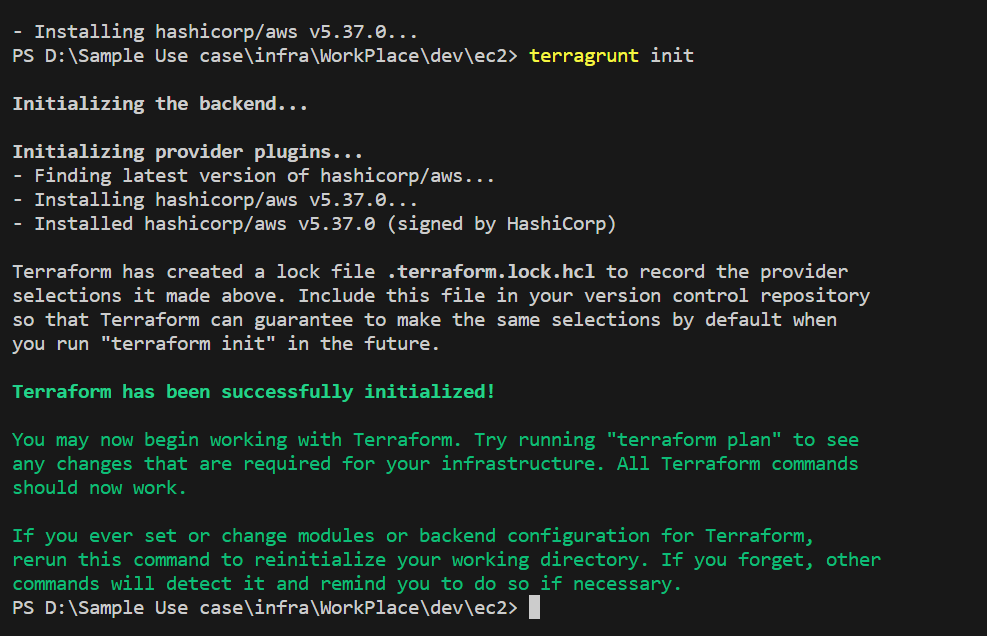
**Infra/terragrunt.hcl**

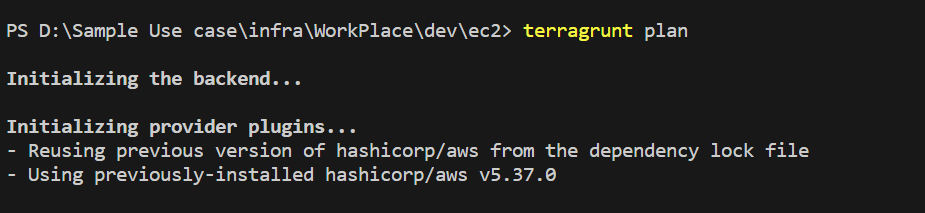
****

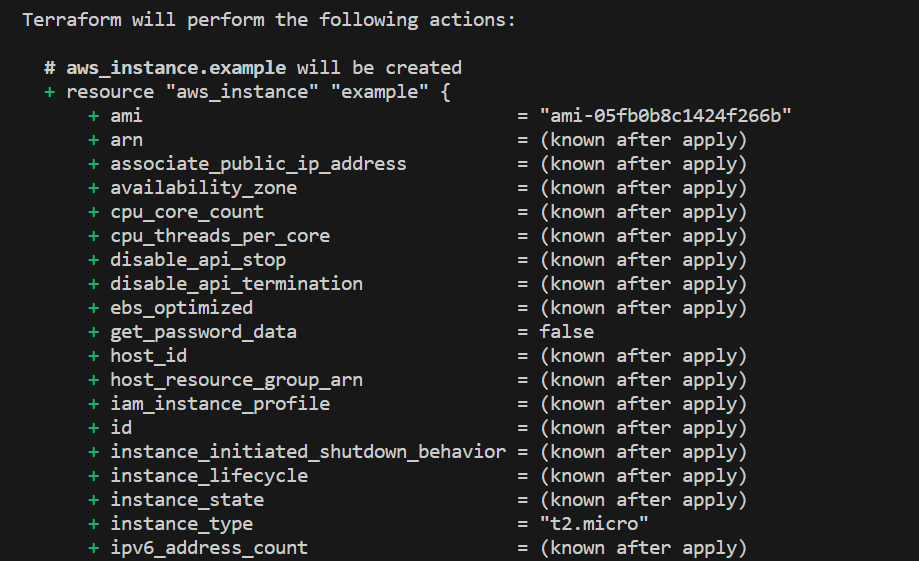
* This .hcl file is created in the root directory for storing our .tfstate file in the remote backend.
* In this method we can easily work with multiple environments with different configuration.

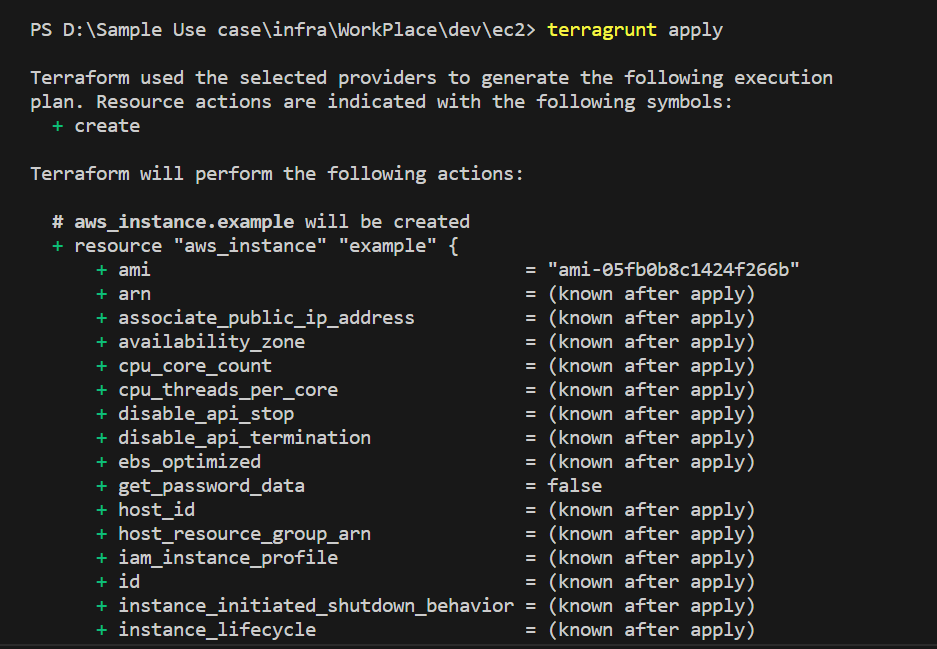
Terragrunt commands also same like Terraform command:

* terragrunt init -> equivalent to [terraform init](https://www.env0.com/blog/terraform-init)
* terragrunt plan -> equivalent to [terraform plan](https://www.env0.com/blog/terraform-plan)
* terragrunt apply -> equivalent to terraform apply
* terragrunt output -> equivalent to terraform output
* terragrunt destroy -> equivalent to terraform destroy









* In this method we can only change the path of directory and use terragrunt commands.
* Different directory contains different config and create resources.

