**1. Write about different modules of Terraform.**

**2. Difference between Root Module and Child Module.**

**3. Are modules and Namespaces being same? Justify your answer for both Yes/No**

1. Root Modules:

Root modules are the top-level Terraform configuration files that define the infrastructure for an entire project. These modules typically contain multiple child modules, resource definitions, and variable declarations.

Child Modules:

Child modules are reusable building blocks that encapsulate a specific set of resources and configurations. These modules can be called from within a root module to instantiate the resources defined within them. Child modules can be reused across different projects and provide a standardized way of defining resources.

Provider Modules:

Provider modules are used to interact with specific cloud providers such as AWS, Google Cloud Platform, or Microsoft Azure. These modules contain the necessary code to interface with the provider's API and create, update, or delete resources. Provider modules can be shared across different projects and enable users to switch between different cloud providers without having to change their code.

Utility Modules:

Utility modules contain common functionalities that are used across multiple projects. These modules can include functions, variables, and templates that can be reused across different modules and projects.

Testing Modules:

Testing modules are used to validate and test the infrastructure code before it is deployed. These modules can be used to perform unit testing, integration testing, and functional testing on the Terraform code.

1. A root module is the top-level Terraform configuration file that defines the infrastructure for an entire project. This module typically contains multiple child modules, resource definitions, and variable declarations. The root module is responsible for coordinating the resources defined within it and is the entry point for executing Terraform commands.

On the other hand, a child module is a reusable building block that encapsulates a specific set of resources and configurations. These modules can be called from within a root module to instantiate the resources defined within them. Child modules can be reused across different projects and provide a standardized way of defining resources.

1. No, modules and namespaces are not the same.

Modules in Terraform are reusable components that encapsulate a set of resources, configurations, and dependencies that can be easily shared and reused across different projects. Modules allow you to abstract and modularize your infrastructure code, making it more maintainable and scalable. Modules can contain variables, inputs, outputs, and resources.

On the other hand, namespaces are a programming concept that allows you to organize code and prevent naming conflicts. In programming, a namespace is a container that holds a set of identifiers, such as variable names, function names, or class names. Namespaces allow you to use the same name for different identifiers, as long as they are in different namespaces.