Chapter1: Introduction to Terraform

* What is terraform?
* What problem it solves?
* How to install terraform?

Chapter2: Making hand Darity with terraform

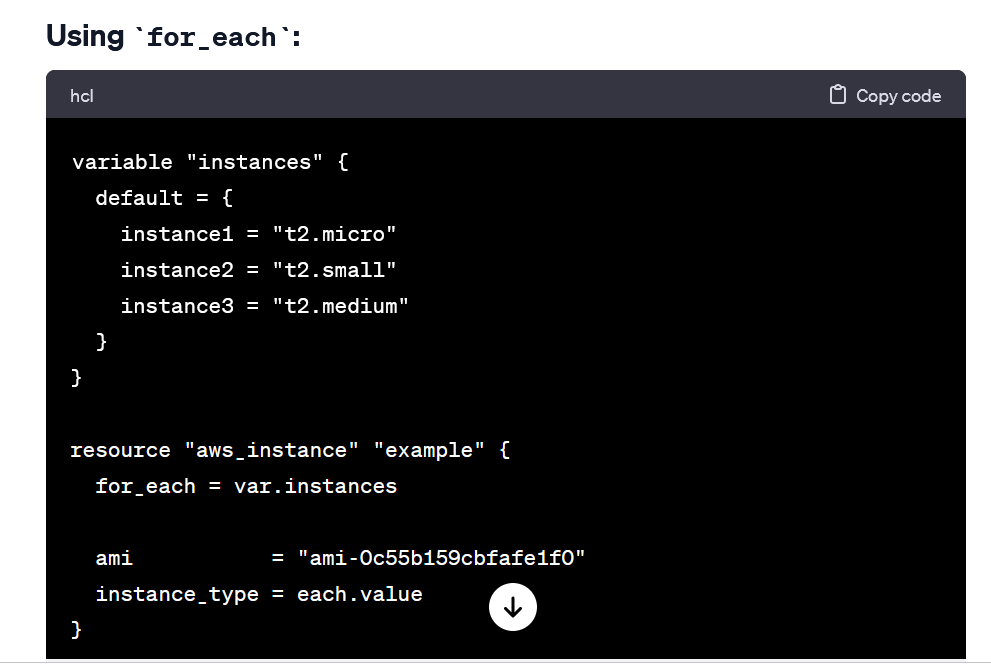
* Provider: Terraform providers help in creating infrastructure in Cloud/Hypervisors or other types. Each Providers will have Resource Types and Data Sources Added
* Resource: This represents the infrastructure element which you want to create.
* Argument: Inputs passed to the Resource are referred as arguments
* Attribute: Outputs are referred as Attributes
* Data Source: Terraform allows to fetch information from provider by a special kind of resource called as Data Source
* EC2 creation with Terraform.
* Terraform init, apply and validate.
* **inconsistent dependency lock file Error**

Chapter3: Deep drive into terraforms init > What is Terraform init, and how it works? The terraform init command initializes a working directory containing Terraform configuration files.It download executable file.

terraform init is a command in Terraform that initializes a working directory containing Terraform configuration files. This command performs several important tasks to set up your Terraform environment:

1. **Backend Initialization:**
   * It initializes the backend, which is the location where Terraform stores its state files. The state file is a critical part of Terraform's operation, as it keeps track of the resources that Terraform manages.
   * The backend can be local, remote (e.g., stored in an object storage service like AWS S3, Azure Storage, etc.), or even a Terraform Enterprise backend.
2. **Plugin Installation:**
   * It downloads and installs the necessary provider plugins specified in your Terraform configuration. Providers are responsible for interacting with APIs of specific infrastructure platforms (e.g., AWS, Azure, Google Cloud).
   * Plugins are downloaded from the official Terraform Registry or a custom provider registry.
3. **Module Installation:**
   * It downloads and installs any modules referenced in your Terraform configuration. Modules are reusable sets of Terraform configuration stored in separate directories.
4. **Initialization of Local Settings:**
   * It initializes local settings and sets up working directories, including creating the .terraform directory where plugins and other local configuration are stored.

Chapter4: Passing runtime argument with terraform. terraform apply -var="ntier\_cidr=192.168.0.0/16" -var='ntier\_subnet\_cidrs=[\"192.168.0.0/24\", \"192.168.1.0/24\", \"192.168.2.0/24\"]'

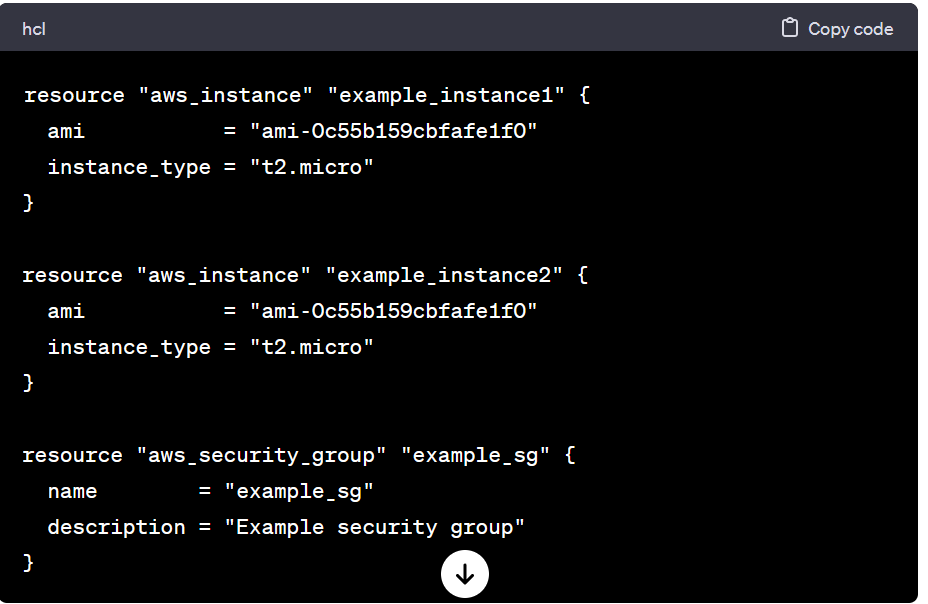
Chapter5: Terraform loop

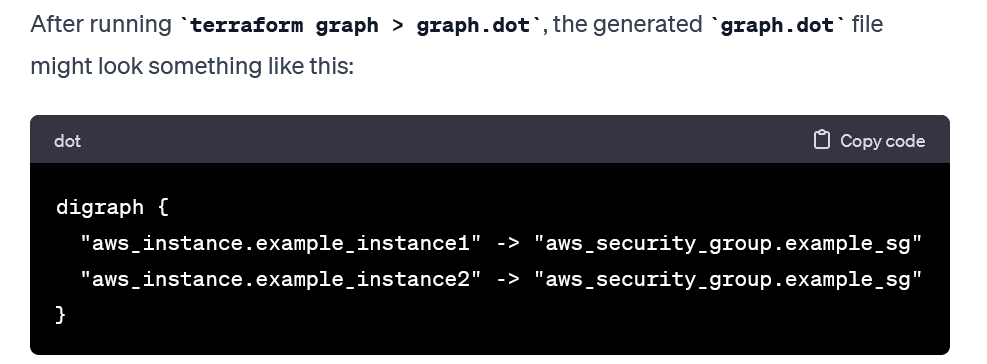
Chapter6: Terraform statefile, console and graph

The state file is used to map real-world resources to your configuration and to store metadata about the resources. It's essential for Terraform to understand the existing state of your infrastructure so that it can plan and apply changes effectively.

The terraform console command is a Terraform CLI (Command-Line Interface) command that allows you to interact with Terraform expressions in an interactive console.

The terraform graph command is a Terraform CLI (Command-Line Interface) command that generates a visual representation of the Terraform resources and their dependencies





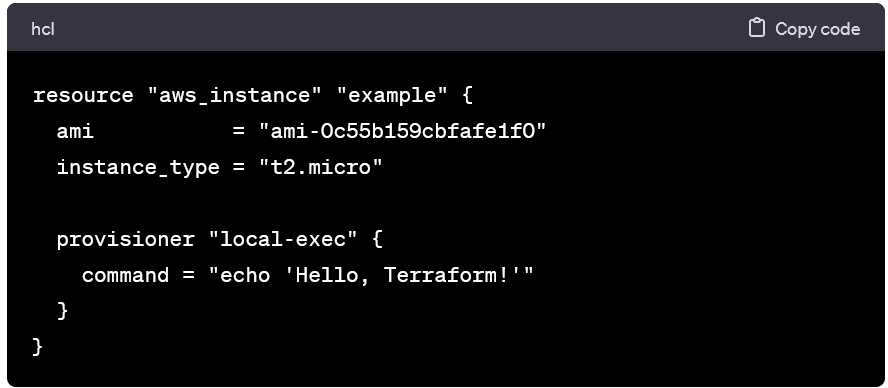
Chapter7: taint and untaint in terraform Terraform has a taint command that informs terraform that a particular object has become degraded or damaged and during the next execution of terraform replace it.

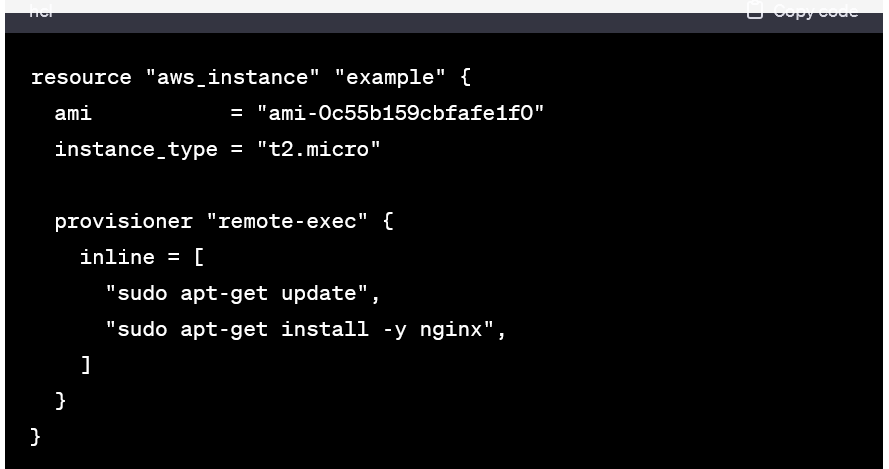
Chapter8: Terraform module’s

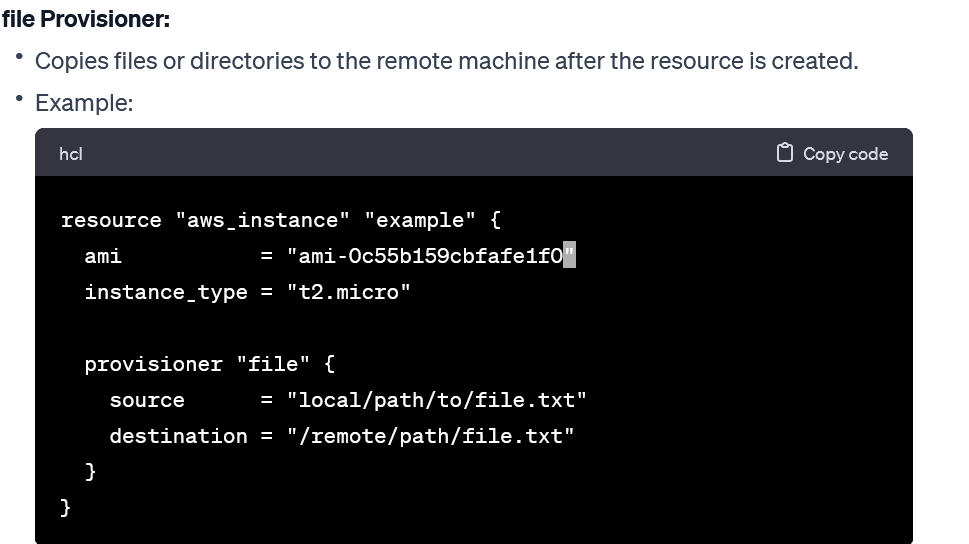
Terraform modules encapsulate reusable infrastructure configurations, enhancing code organization and promoting consistency in deployments.

Chapter9: Terraform provisioning and null resource.

In Terraform, provisioners are used to perform actions on the remote resource instances after they are created. Provisioners can be helpful for tasks such as installing software, running configuration scripts, or performing other actions on the provisioned infrastructure.







Chapter10: **Terraform Backend:** A backend defines where terraform stores its state data files.

Chapter11: Packer

Convergence and idempotence behaviour:

In convergence approach we focus on desire state from current state.

Idempotence: it’s like if we run same command multiple time, it will not impact anything.

