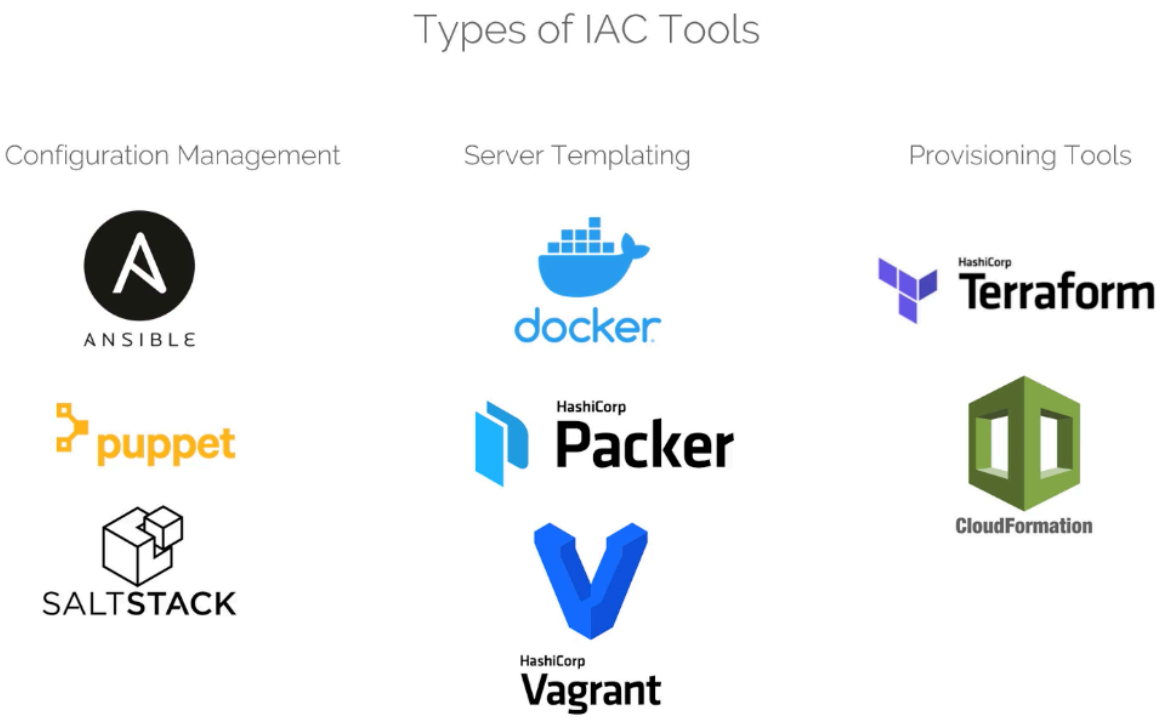
Terraform from scratch:
~~~~~~~~~~~~~~~~~~~~~~~~



Some interview questions:
~~~~~~~~~~~~~~~~~~~~~~~~~

Terraform:
~~~~~~

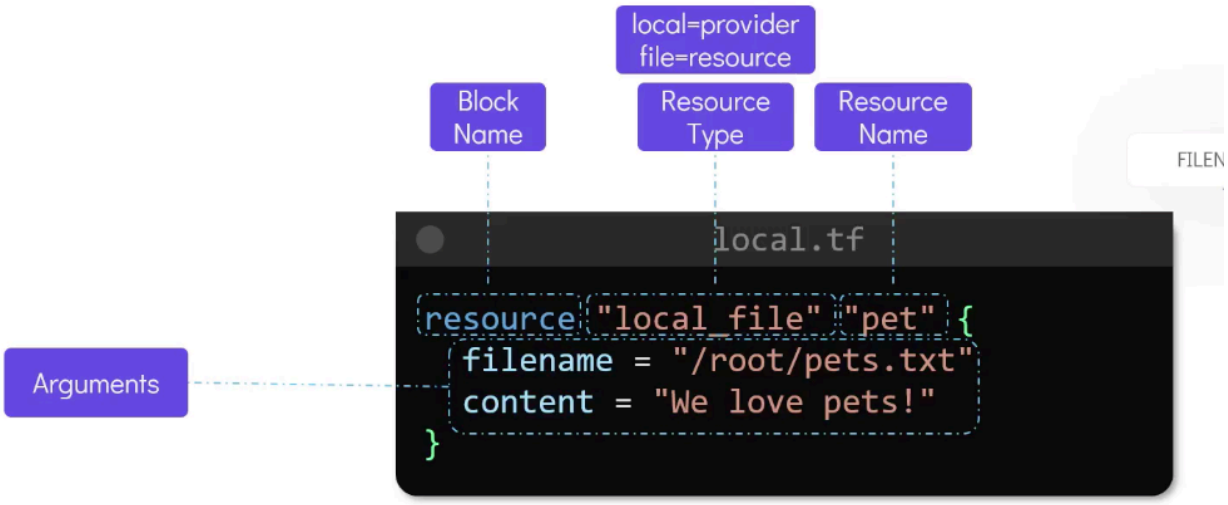
1. Explain the main components of Terraform.
2. What is the purpose of Terraform's state file? How does it manage resources and prevent conflicts in a team setting?
3. Can you describe the Terraform workflow ?
4. How does Terraform handle dependencies between resources?
5. What is the purpose of the "terraform.tfvarsfile" and how does it relate to variable definitions in Terraform configurations?
6. Explain the difference between Terraform's provider and resource blocks.
7. What are Terraform modules, and why are they used?
8. How does Terraform manage updates to infrastructure without causing downtime?
9. Describe the remote back end in Terraform and its significance in a collaborative environment.
10. How does Terraform ensure idempotency, and what is its significance?
11. What are Terraform workspaces? How are they utilized?
~~~~~~

12. What strategies can you use for managing secrets or sensitive information in Terraform configurations?
13. Explain the concept of Terraform variables and outputs. How are they used, and what is their significance?
14. Discuss Terraform state locking mechanisms and their importance in a multi-user environment.
15. Discuss the differences between Terraform's count and for_each meta-arguments when defining resources.

https://github.com/bregman-arie/devops-exercises?tab=readme-ov-file

~~~~~~

Basics:

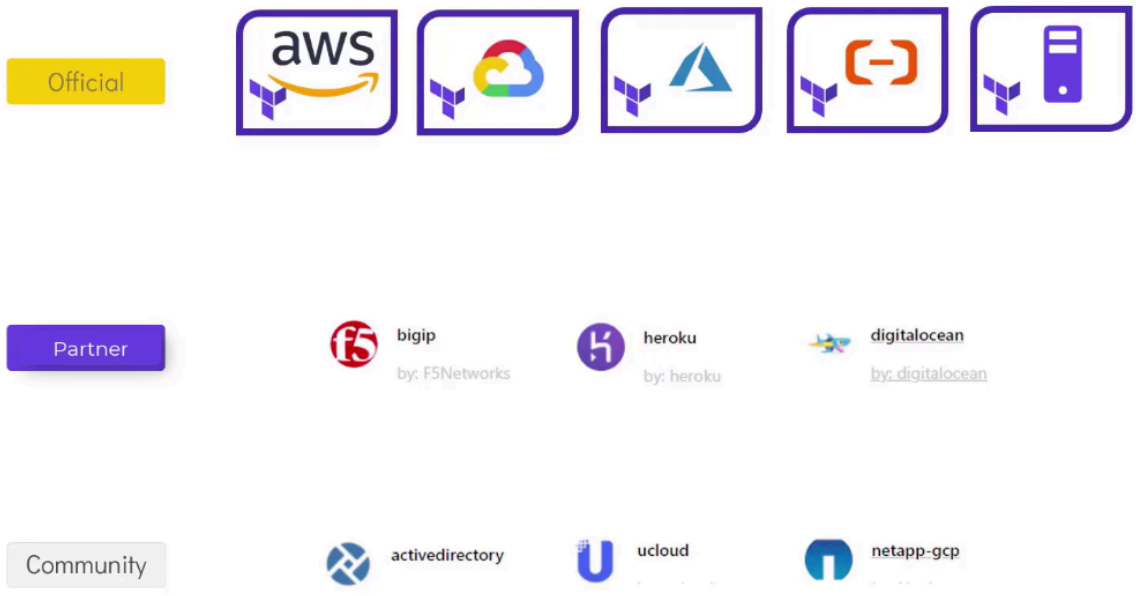


Playing with provider local:

```
resource "local_file" "file1" {
   filename = "root/file.txt"
   content = "hello using local provider"
   file_permission = 0770
}
```
~~~~~~

```
resource "local_sensitive_file" "sensitive_file1" {
   filename = "/root/sensitive_file1.txt"
   content = "Since I am using sensitive_file resource type, data will not be shown
while doing terraform apply"
   file_permission = 0770
 }
```

Using Terraform Providers

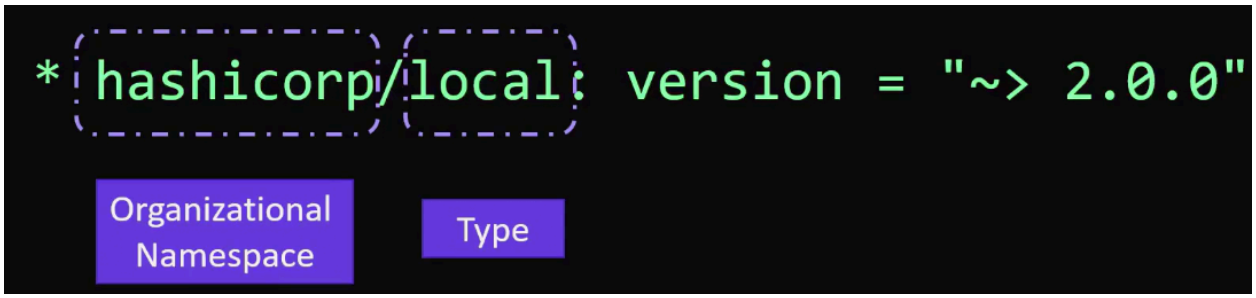


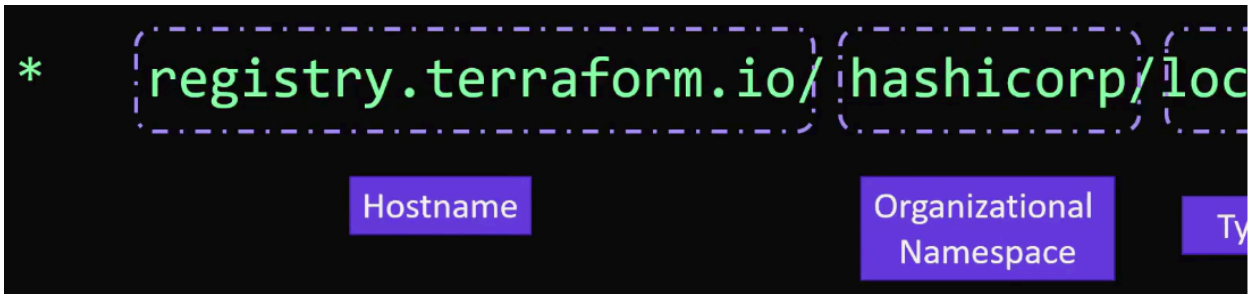
When you run ‘terraform init’ , it will initialize the required plugins basing on the .tf files content present in the current directory.

```
$ terraform init
Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/local...
- Installing hashicorp/local v2.0.0...
- Installed hashicorp/local v2.0.0 (signed by HashiCorp)
```

```
>_
$ ls /root/terraform-local-file/.terraform
plugins
```





When you run ‘terraform initi’ , it downloads the latest plugins so it may break your existing code.

How to avoid that ? it will be discussed in later sessions?