True / False:

1. You can migrate the Terraform backend but only if there are no resources currently being managed.  
   ANS: FALSE
2. Workspaces provide similar functionality in the open-source and Terraform Cloud versions of Terraform.

ANS: FALSE

1. Similar to Terraform OSS, you must use the CLI to switch between workspaces when using Terraform Cloud workspaces.

ANS: FALSE

1. Provisioners should only be used as a last resort.

ANS: **TRUE**

1. State is a requirement for Terraform to function.

ANS: **TRUE**

1. Multiple providers can be declared within a single Terraform configuration file.

ANS: **TRUE**

1. By default, the terraform destroy command will prompt the user for confirmation before proceeding.  
   ANS: **TRUE**
2. When using the Terraform provider for Vault, the tight integration between these HashiCorp tools provides the ability to mask secrets in the state file.  
   ANS: **FALSE**
3. The terraform plan -refresh-only command is used to create a plan whose goal is only to update the Terraform state to match any changes made to remote objects outside of Terraform.  
   ANS: **TRUE**
4. Each Terraform workspace uses its own state file to manage the infrastructure associated with that particular workspace.  
   ANS: **TRUE**
5. Rather than use a state file, Terraform can inspect cloud resources on every run to validate that the real-world resources match the desired state.  
   ANS: **FALSE**
6. Using the latest versions of Terraform, terraform init cannot automatically download community providers.  
   ANS: **FALSE**
7. A terraform plan is a required step before running a terraform apply?  
   ANS: **FALSE**
8. Any sensitive values referenced in the Terraform code, even as variables, will end up in plain text in the state file.  
   ANS: **TRUE**
9. Terraform is designed to work only with public cloud platforms, and organizations that wish to use it for on-premises infrastructure (private cloud) should look for an alternative solution.  
   ANS: **FALSE**
10. A remote backend configuration is required for using Terraform.  
    ANS: **FALSE**
11. A main.tf file is always required when using Terraform?  
    ANS: **FALSE**
12. In both Terraform OSS and Terraform Cloud, workspaces provide similar functionality of using a separate state file for each workspace.  
    ANS: **TRUE**
13. When developing Terraform code, you must include a provider block for each unique provider so Terraform knows which ones you want to download and use.  
    ANS: **FALSE**
14. You can continue using your local Terraform CLI to execute terraform plan and terraform apply operations while using Terraform Cloud as the backend.  
    ANS: **TRUE**
15. min, max, format, join, trim, and length are examples of different expressions in Terraform.  
    ANS: **FALSE**
16. Terraform can only manage dependencies between resources if the depends\_on argument is explicitly set for the dependent resources.  
    ANS: **FALSE**
17. If you have properly locked down access to your state file, it is safe to provide sensitive values inside of your Terraform configuration.  
    ANS: **TRUE**
18. When using Terraform Cloud, committing code to your version control system (VCS) can automatically trigger a speculative plan.  
    ANS: **TRUE**
19. terraform validate will validate the syntax of your HCL files.  
    ANS: **TRUE**
20. The terraform graph command can be used to generate a visual representation of a configuration or execution plan.  
    ANS: **TRUE**
21. Official Terraform providers and modules are owned and maintained by HashiCorp.  
    ANS: **TRUE**
22. In most cases, you can move Terraform state between supported backends at any time, even after running your first terraform apply.  
    ANS: **TRUE**
23. Input variables that are marked as sensitive are NOT written to Terraform state.  
    ANS: **FALSE**
24. Infrastructure as code (IaC) tools allow you to manage infrastructure with configuration files rather than through a graphical user interface.  
    ANS: **TRUE**
25. Under special circumstances, Terraform can be used without state.  
    ANS: **FALSE**
26. A provider block is required in every configuration file so Terraform can download the proper plugin.  
    ANS: **FALSE**
27. Running a terraform apply will fail if you do not run a terraform plan first.  
    ANS: **FALSE**
28. When referencing a module, you must specify the version of the module in the calling module block.  
    ANS: **FALSE**
29. In order to use the terraform console command, the CLI must be able to lock state to prevent changes.  
    ANS: **TRUE**
30. In Terraform OSS, workspaces generally use the same code repository while workspaces in Terraform Enterprise/Cloud are often mapped to different code repositories.  
    ANS: **TRUE**
31. You can use a combination of Terraform Cloud's cost estimation feature and Sentinel policies to ensure your organization doesn't apply changes to your environment that would result in exceeding your monthly operating budget.  
    ANS: **TRUE**

Numbers:

1. In Terraform Cloud, a workspace can be mapped to how many **VCS repos**?  
   ANS: **1**
2. In order to reduce the time it takes to provision resources, Terraform uses parallelism. By default, how many resources will Terraform **provision concurrently** during a terraform apply?  
   ANS: **10**
3. When writing Terraform code, how many spaces between each **nesting level** does HashiCorp recommends that you use?  
   ANS: **2**

Dear Karthik,

I wanted to take a moment to extend my heartfelt appreciation for your exceptional performance and dedication to our team during IntlSOS project.

Your exemplary work in successfully transitioning AWS accounts to Orange has been nothing short of exceptional. Your attention to detail, meticulous planning, and strategic approach has played a pivotal role in ensuring a smooth and seamless transition. Your commitment to excellence was evident throughout the process, and it has had a positive impact on our entire team. Your efforts in the Backup (Gp3 to St1) were truly remarkable. Your sound technical knowledge and expertise were instrumental in identifying and resolving complex challenges along the way. Your go-get attitude is infectious and inspiring. You consistently show a strong determination to excel and achieve the best results possible. Your enthusiasm and drive have set a benchmark for the entire team and have motivated us all to push beyond our limits.

Warm regards,

Madhu Sudhana Reddy Indukuri

AWS & Azure Architect