

TERRAFORM INTERVIEW QUESTIONS PDF

1. What is the full form of IAC in the context of Terraform?

- The full form of the term IAC is "Infrastructure as Code".
- IaC refers to a scheme whereby developers can run and provision the computer data center's mechanically instead of getting into a physical process.

2. What are some major competitors of Terraform?

- Packer
- Cloud Foundry
- Ansible
- Kubernetes

3. What do you understand by Terraform Backends? What are the most recommended Backends we should use?

- If the new users are still learning how to use Terraform, the default "local" backend is most recommended, which requires no configuration.
- For the expert users, if they or their team are using Terraform to manage meaningful infrastructure, the most recommended backend is the "remote" backend with Terraform Cloud or Terraform Enterprise.

4. What do you understand by Oracle Cloud Infrastructure?

- Oracle Cloud Infrastructure is a cloud computing service offered by Oracle Corporation.
- It provides storage, servers, applications, services, and networks through a global network of managed data centers by Oracle Corporation.

5. Why is Terraform used for DevOps?

- Terraform is preferred to use for DevOps because it facilitates us to manage infrastructure as code.
- The infrastructure as code is the foundation for DevOps practices such as continuous integration, version control, continuous deployment, and code review.

6. What do you understand by Terraform providers?

- Providers are plugins on that Terraform relies to interact with cloud providers, SaaS providers, and other APIs.
- The Terraform configurations have to declare which providers they require so that Terraform can install and use them.

7. What are the most important features of Terraform?

- **Infrastructure as Code:** The high-level configuration language of Terraform is used to describe the infrastructure in declarative configuration files that are human-readable.

- **Execution Strategies:** Before making any infrastructure modifications, Terraform develops an execution plan to describe what it will do and asks for your agreement.
- **Graph of Resources:** Terraform develops or alters non-dependent resources while simultaneously building a resource graph.
- **Automation of Change:** Terraform can be used to automate the application of complex changesets to your infrastructure with little to no human intervention.

8. What is the use of Terraform provider?

- A provider implements every resource type; without providers, Terraform can't manage any infrastructure.
- Terraform providers configure a specific infrastructure platform (either cloud or self-hosted).
- Terraform providers can also offer local utilities for tasks such as generating random numbers for unique resource names

9. What do you understand by Terraform in AWS?

- In AWS, Terraform is an infrastructure as a code tool.
- It facilitates us to store our Amazon Web Services infrastructure and produce an update, and it is very similar to AWS Cloud Formation

10. What are the biggest competitors of Terraform?

- Kubernetes
- Turbonomic
- CloudBolt
- Azure Management Tools
- Morpheus
- CloudHealth
- Apptio Cloudability, more

11. What is the usage of Terraform init?

- It is used for Plugin Installation.
- It is used for Backend Initialization.
- You can safely run this command multiple times.

12. What are some Built-in provisioners available in Terraform?

- File Provisioner
- Chef Provisioner
- Habitat Provisioner
- Puppet Provisioner
- Local-exec Provisioner
- Remote-exec Provisioner
- Salt-masterless Provisioner

13. What are the usages of Terraform CLI?

- The Terraform CLI stands for Terraform Command Line Interface. It is used to manage infrastructure and interact with Terraform state, configuration files, providers, etc.

14. What is the usage of provisioners in Terraform?

- In Terraform, provisioners are used to preparing servers or other infrastructure objects for service.
- They are used on the local machine as well as on the remote machines.

15. What is Terraform Cloud for Business?

- Terraform Cloud for Business is very much similar to Terraform Cloud and it also uses the same hosted environment as Terraform Cloud but with additional features for larger teams.
- It gives some additional features such as single sign-on, audit logging, and the ability to Terraform on-premise resources from the Cloud,more.

16. What is the null resource in Terraform?

- The null_resource is used to implement a standard resource library, but no further action is taken.
- The triggers argument allows an arbitrary set of values that will cause the replacement of resources when changed.

17. What are some basic CLI commands?

- terraform init
- terraform destroy
- terraform validate
- terraform apply
- terraform plan
- terraform refresh
- terraform graph

18. What is ".terraform" directory?

- The ".terraform" directory is a local cache where Terraform retains some files required for subsequent operations against this configuration.
- Its contents are not intended to be included in version control.

19. What do you understand by Terraform backend?

- The Terraform backend is used to specify how an operation is executed and how the state is loaded.
- It uses the 'local' backend by default. The abstraction enables remote execution, non-local file state storage, etc.

20. What can you do to ignore duplicate resource errors during Terraform apply?

- You can delete those resources from Terraform code to stop its management.
- You can do a Terraform import of the resource and remove the code trying to recreate them.
- You can delete those resources from the cloud provider (API) and recreate them using Terraform.

21. What do you understand by Terraform init?

- Terraform init is a control used to initialize an operational index containing Terraform pattern files. This control is the first command that should be run after writing the new Terraform design and can be looped multiple times.

22. Is there any way to lock the Terraform Module Versions?

- There is a way to lock the Terraform Module Versions. If the Terraform module registry is being used as a source, then the 'version' attribute can be used in the module in a configuration file of Terraform.

23. What are the version controls supported by Terraform?

- Azure DevOps Services
- Azure DevOps Server
- Bitbucket Server
- Bitbucket Cloud
- Gitlab EE and CE
- Gitlab.com
- GitHub Enterprise
- GitHub.com
- GitHub.com (OAuth)

24. What is the Remote Backend in Terraform?

- In Terraform, the Remote Backend is an extension used to state the status of Terraform and run the Terraform cloud operations.
- The backend operation consists of several commands to carry out all the operations of Terraform.
- The remote function can work on a single cloud workspace to access multiple fronts for executing remote applications like a plan or application.

25. How can we create a dependency between modules in Terraform?

- Terraform, the dependencies between modules are generally created automatically by references, rather than explicitly using depends_on. In Terraform, there is no way to use depends_on variables.

26. What are the main reasons for choosing Terraform for DevOps?

- It provides amazing support to almost all the popular cloud providers like AWS, Azure, GCP, DigitalOcean, etc.
- It can easily manage the configuration of an immutable (dynamic) infrastructure.

- It provides an immutable infrastructure where configuration changes smoothly.
- It works on HCL (HashiCorp Configuration Language), which is very easy to learn and understand.
- It is easily portable from one provider to another.

27. What is the full form of Terraform HCL?

- The full form of Terraform HCL is Terraform HashiCorp Configuration Language.
- It is a configuration language built by HashiCorp Corporation. I
- t is used to build a structured configuration language that is both human and machine-friendly for use with command-line tools but specifically targeted towards DevOps tools, server.

28. What is Terraform D?

- Terraform D is a plugin used on most in-service systems and Windows. Terraform init by default searches next directories for plugins.

29. Difference between Terraform and CloudFormation?

Terraform	Cloudformation
Terraform can work with various cloud providers such as AWS, GCP, Azure	Cloudformation may be complicated if your setup requires several cloud installations. It is best for AWS resources such as EC2, S3
Terraform saves the state of the infrastructure on a virtual machine or a remote computer.	The state of Cloudformation is managed out-of-the-box by CloudFormation, which prevents conflicting updates.

30. What are the modules in Terraform?

- A Terraform module is a set of numerous resources used jointly.
- It is a single directory containing Terraform configuration files.
- A simple arrangement with a single directory having one or more files can be referred to as a module.