

Terraform Interview Questions

1. Define IAC?
2. What are Terraform Providers?
3. Explain the basic structure of a Terraform configuration file
4. What is a Terraform state file?
5. How do you initialize a Terraform configuration?
6. What is the purpose of the terraform plan command?
7. Explain the concept of a "resource" in Terraform.
8. What is the purpose and functionality of the null_resource in Terraform?
9. How can you manage different environments (e.g., dev, prod) using Terraform?
10. What are Terraform modules and how do they help?
11. Explain the difference between terraform apply and terraform plan -out.
12. What are remote backends in Terraform?
13. How do you handle secrets in Terraform?
14. What is Terraform's taint command used for?
15. What role do provider aliases play in a multi-account setup?
16. How does DynamoDB help in state locking, and why is it needed?
17. What is the difference between terraform destroy and terraform state rm
18. What is the difference between terraform refresh and terraform apply in such a situation?
19. How do you prevent Terraform from destroying and recreating resources when the count changes?
20. Explain the use of count and for_each in Terraform.
21. How can you create multiple EC2 instances dynamically in Terraform?
22. How do you import existing infrastructure into Terraform?
23. What exactly is meant by a "Private Module Registry"?
24. Can Terraform deploy resources across multiple providers?
25. What are Terraform's built-in functions and how can they be used?
26. How do you manage Terraform configurations in a CI/CD pipeline?
27. How does Terraform handle dependencies between resources?

28. What are Terraform workspaces and how do they differ from modules?
29. How do you debug and troubleshoot Terraform configurations?
30. Explain the use of terraform state commands.
31. What is the concept of state file locking, and how does it work?
32. How do Terraform and Ansible differ in terms of their functionalities and use cases?
33. Your Terraform configuration deploys an EC2 instance. However, a team member manually changed the instance type in the AWS console. How can you detect and reconcile this change in Terraform?
34. A junior developer mistakenly removed an S3 bucket from the Terraform configuration file. On running terraform apply, Terraform detects it as a deletion. How can you prevent accidental deletions?
35. Your Terraform deployment failed because a security group needed for an EC2 instance was not created in time. How can you ensure Terraform manages dependencies correctly?
36. A Terraform state file became corrupted due to an unexpected system failure. How do you recover it?
37. What is the difference between Terraform and Ansible?
38. How does Terraform compare to CloudFormation for AWS provisioning?
39. What are the best practices for structuring Terraform configurations?