

# 40 Terraform Interview Questions to Assess Candidates

## Questions

---

1. What is Terraform, and why would you use it?
2. Can you explain what a Terraform provider is and give an example?
3. What is a Terraform state file, and why is it important?
4. How does Terraform handle resource dependencies?
5. What is a Terraform module, and how do you use it?
6. Can you describe the difference between a Terraform 'plan' and 'apply' command?
7. What are Terraform workspaces, and when would you use them?
8. How do you ensure the security of sensitive data in Terraform configurations?
9. What are some common challenges you might face when using Terraform, and how would you address them?
10. Can you give an example of a real-world scenario where you successfully used Terraform?
11. What are the benefits of using Terraform over traditional configuration management tools like Ansible or Chef?
12. How do you manage Terraform configurations for different environments, such as development and production?
13. Can you explain the purpose and structure of a Terraform variable file?
14. What is the 'terraform init' command used for, and why is it necessary?
15. How do you handle version control for Terraform configurations?
16. What are remote backends in Terraform, and why would you use them?
17. Can you describe the purpose of the 'terraform fmt' command?
18. How would you approach debugging a Terraform configuration that is not working as expected?
19. What is the difference between 'terraform destroy' and removing resources manually?
20. How do you use output variables in Terraform, and why are they important?
21. How would you explain the concept of infrastructure as code to someone unfamiliar with it?
22. How do you manage and organize Terraform files for a large project?
23. What are some best practices for writing Terraform configurations?
24. How does Terraform ensure idempotency in infrastructure management?
25. What steps would you take to ensure the reliability of your Terraform deployments?
26. How do you handle Terraform state files across different environments or teams?
27. What are the key differences between declarative and imperative approaches in infrastructure management, and where does Terraform fit in?
28. Can you explain the concept of 'immutable infrastructure' and how Terraform supports this approach?
29. What strategies do you use to keep your Terraform codebase maintainable and reusable over time?
30. Can you explain the importance of using version constraints in Terraform provider configurations?
31. How do you ensure your Terraform configurations are consistent across different team members?
32. What methods do you use to document your Terraform infrastructure?
33. How would you implement a testing strategy for your Terraform configurations?
34. In what situations would you choose to use local modules versus remote modules?
35. How do you handle secrets management in Terraform without exposing sensitive information?
36. Can you discuss the trade-offs between using Terraform CLI directly versus using Terraform Cloud or Enterprise?
37. What is the role of a 'terraform workspace' in managing different stages of a project?
38. How do you manage resource lifecycle in Terraform to avoid unintended changes?
39. What are some common pitfalls to avoid when structuring Terraform files for a large project?