Terraform Basics

1. **What is Terraform, and why is it used?**
2. **How does Terraform differ from other Infrastructure as Code (IaC) tools like Ansible, CloudFormation, and Pulumi?**
3. **What are the main components of Terraform?** *(Provider, Resources, Modules, State, etc.)*
4. **Explain the Terraform workflow (init, plan, apply, destroy).**
5. **What is the Terraform configuration file format, and what file extensions does it use?**

Terraform State Management

6. **What is Terraform state, and why is it needed?**
7. **Where can Terraform store its state file?**
8. **What is the difference between `terraform.tfstate` and `terraform.tfstate.backup`?**
9. **What are the risks of storing Terraform state locally, and how can they be mitigated?**
10. **How do you enable remote backend storage for Terraform state?**
11. **What happens if two people run Terraform at the same time?**
12. **What is state locking in Terraform, and how does it work?**
13. **How do you perform a Terraform state import?**
14. **How can you remove a resource from the Terraform state without destroying it?** *(Hint: `terraform state rm`)*

Terraform Provisioning & Deployment

15. **What is the difference between `terraform apply` and `terraform destroy`?**
16. **How does Terraform handle dependencies between resources?**
17. **What is a `terraform refresh` command used for?**
18. **How do you taint a resource in Terraform, and when would you use it?**
19. **How can you rollback changes in Terraform if something goes wrong?**
20. **How do you apply only specific resources from your Terraform configuration?**
21. **How do you dry-run Terraform before making actual changes?**
22. **How does Terraform handle failures during resource creation?**
23. **What happens if a Terraform apply is interrupted?**
24. **Can Terraform be used with existing infrastructure?** *(Hint: `terraform import`)*

Terraform Modules & Reusability

25. **What are Terraform modules, and why should you use them?**
26. **How do you create a reusable Terraform module?**

27. **What is the difference between a root module and a child module?**
28. **How do you reference a module inside a Terraform configuration?**
29. **What is the purpose of `terraform get` and `terraform init` in the context of modules?**
30. **How do you pass variables to a Terraform module?**
31. **Can Terraform modules have dependencies?**
32. **What are some best practices for structuring Terraform modules?**
33. **How do you version control Terraform modules?**

Terraform Variables & Outputs

34. **What are Terraform variables, and how are they used?**
35. **What is the difference between input and output variables in Terraform?**
36. **How do you define and use variables in Terraform?**
37. **What is the difference between `var` and `locals` in Terraform?**
38. **What is a Terraform `output` variable, and how is it used?**
39. **How do you reference a variable from another module?**
40. **What are Terraform environment variables, and how do you use them?**
41. **How can you assign default values to variables in Terraform?**
42. **What are sensitive variables in Terraform, and how do you handle them?**
43. **How do you pass variables from a `.tfvars` file?**
44. **How can you use `terraform console` to inspect values in your configuration?**

Terraform Providers & Backends

45. **What are Terraform providers, and why are they needed?**
46. **How do you define a provider in Terraform?**
47. **What is a provider alias in Terraform, and when would you use it?**
48. **How do you update or upgrade a Terraform provider?**
49. **How do you authenticate with cloud providers in Terraform?**
50. **What are Terraform backends, and why are they important?**
51. **What is the difference between local and remote backends in Terraform?**
52. **How do you migrate from a local backend to a remote backend?**
53. **Can you use multiple providers in a single Terraform configuration?**
54. **How do you configure AWS credentials for Terraform?**
55. **How do you use Terraform with Azure/AWS/GCP?**

Terraform Workspaces & Environments

56. **What is a Terraform workspace, and why would you use it?**
57. **How do you create and switch Terraform workspaces?**
58. **What are the differences between Terraform workspaces and Git branches?**
59. **How do you manage different environments (dev, stage, prod) in Terraform?**

60. **What are some best practices for handling multiple environments in Terraform?**

Terraform Best Practices & Security

61. **How do you manage secrets in Terraform?**
62. **What are some security best practices for Terraform?**
63. **How do you scan Terraform code for security vulnerabilities?**
64. **How do you ensure Terraform code is following best practices?**
65. **What are some Terraform linting tools you use?**
66. **How do you integrate Terraform with CI/CD pipelines?**
67. **What are some best practices for Terraform state management?**
68. **How do you handle drift detection in Terraform?**
69. **How do you enforce policies in Terraform?** *(Hint: `Sentinel`, `OPA`)*

Terraform Advanced Topics

70. **What is Terraform Cloud, and how is it different from Terraform CLI?**
71. **What is the purpose of Terraform Sentinel?**
72. **What is the difference between `count` and `for_each` in Terraform?**
73. **How do you dynamically create multiple resources in Terraform?**
74. **What is the `terraform fmt` command used for?**
75. **How does Terraform handle resource dependencies automatically?**
76. **How do you handle a situation where Terraform fails to destroy resources?**
77. **What is a Terraform plan file, and how do you use it?**
78. **How does Terraform handle API rate limits?**
79. **How do you use the `terraform graph` command?**
80. **How do you recover from accidental state deletion in Terraform?**

Scenario-Based Questions

81. **You have a Terraform project deployed in AWS. Someone manually changed the security group in the AWS console. How do you reconcile this in Terraform?**
82. **You need to deploy the same infrastructure across multiple AWS accounts. How do you structure your Terraform code?**
83. **Your Terraform apply command fails due to a missing dependency. How do you troubleshoot this?**
84. **Your team accidentally deleted the Terraform state file. What steps do you take to recover it?**
85. **You need to migrate Terraform state from local to S3 with DynamoDB for state locking. How would you do it?**
86. **You are using Terraform in a CI/CD pipeline. How do you securely pass AWS credentials?**
87. **How do you handle Terraform in a GitOps workflow?**

88. **You need to modify an existing Terraform resource without destroying it. How do you approach it?**
89. **Your Terraform state is corrupted. How do you troubleshoot and fix it?**
90. **A new Terraform provider version is available. How do you upgrade it safely?**