1. **A user want to upgrade the terraform plugin, how to upgrade it ?**
2. Terraform plan --upgrade
3. Terraform fmt
4. Terraform init –upgrade.
5. Aws configure.

**2. Can you specify multiple dependencies using the "depends\_on" attribute in Terraform?**

A) Yes, using a comma-separated list

B) No, only a single dependency can be specified

C) Yes, using a list syntax

D) No, Terraform automatically infers dependencies.

**3. What is the purpose of the depends\_on argument in a data source block?**

A) It specifies the order of resource creation

B) It defines the data source dependencies

C) It specifies the version of the data source

D) It configures access permissions

Answer: B) It defines the data source dependencies

**4.How can you dynamically generate resource configurations in Terraform?**

A) Using count and for\_each

B) Using dynamic blocks

C) Using modules

D) All of the above

1. **What command is used to remove a specific resource from the state file without destroying the resource?**

A) terraform remove

B) terraform state rm

C) terraform delete

D) terraform state del

**6. A user configured the S3 backend to store the state file for one of their projects. After running terraform apply, the state file was stored. Another user used the same path in the S3 bucket to store the state file for a different project. If this user runs terraform apply, what will happen?**

1. **Delete first user configuration changes**
2. **Update second user state file also in same path**
3. **Second user end will get error like state file already configured**
4. **Second user changes also will updated without any disturbance of first user**

**7.How do you reference a module within a Terraform configuration file?**

A) resource "module" "name"

B) module "name".

C) resource "name"

D) module.resource "name

**8.What file must be included in a directory for it to be recognized as a Terraform module?**

A) module.tf

B) main.tf.

C) variables.tf

D) outputs.tf

**9.The user wrote the Terraform configuration file to create the EC2 instance and deploy the sample application. After the Terraform state file was updated, is the application creation status updated in the state file?**

a. No

b. Yes

**10.How can you dynamically generate resource configurations in Terraform?**

A) Using count and for\_each

B) Using dynamic blocks

C) Using modules

D) All of the above

**11.User want to destroy specific ec2 resource only from local configuration file how to delete it ?**

**a) terraform state rm aws\_instance.example.**

**b) terraform destroy aws\_instance.example**

**c) terraform apply -target=aws\_instance.example**

**d)** **terraform delete -target=aws\_instance.example**

**12. User want to creates specific ec2 resource only from local configuration file how to create it ?**

**a) terraform state apply aws\_instance.example.**

**b) terraform apply aws\_instance.example**

**c) terraform apply -target=aws\_instance.example**

**d) terraform delete -target=aws\_instance.example**

1. **How can you source a module from a Git repository?**

A) source = "git@github.com:user/repo"

B) source = "git::https://github.com/user/repo.git"

C) source = "https::git@github.com/user/repo."

D) source = "github.com/user/repo"

1. **Where is the Terraform state file stored by default?**

A) In a remote S3 bucket

B) In a database

C) Locally in the Terraform configuration directory

D) In the Terraform Cloud

**15.What happens if two users run terraform apply concurrently on the same state file stored in a remote backend?**

A) The second apply command will fail due to state file locking

B) Both commands will run successfully and merge changes

C) The state file will be corrupted

D) Terraform will automatically resolve any conflicts

**16.A user runs terraform init on their server, and per the output, two provider plugins are downloaded:**

1. The. terraform/providers directory in the current working directory
2. The. terraform.d directory in the current working directory
3. /etc/terraform/plugins
4. The. terraform.plugins directory in the current working directory

**17.Both you and a colleague are responsible for maintaining resources that host multiple applications using Terraform CLI. What feature of Terraform helps ensure only a single person can update or make changes to the resources Terraform is managing?**

1. version control
2. 2. local backend
3. provisioners
4. 4. state locking

**18.You have a Terraform configuration file with no defined resources. However, there is   a related state file for resources that were created on AWS. What happens when you run a terraform apply?**

1. Terraform will destroy all of the resources.
2. Terraform will not perform any operations
3. Terraform will produce an error since there are no resources defined
4. Terraform will scan the AWS infrastructure and create a new configuration file based on the state file

**19. Infrastructure as Code (IaC) provides many benefits to help organizations deploy application infrastructure much faster than clicking around in the console. What are the additional benefits of IaC? (select three)**

1. can always be used to deploy the latest features and services
2. creates a blueprint of your data center
3. code can easily be shared and reused
4. eliminates parallelism
5. allows infrastructure to be versioned

**20 You are working with a cloud provider to deploy resources using Terraform. You've added the following data block to your configuration. When the data block is used, what data will be returned?**

data "aws\_ami" "amzlinux2" {

  most\_recent = true

  owners      = ["amazon"]

  filter {

    name   = "name"

    values = ["amzn2-ami-hvm-\*-x86\_64-ebs"]

  }

}

resource "aws\_instance" "vault" {

  ami                         = data.aws\_ami.amzlinux2.id

  instance\_type               = "t3.micro"

  key\_name                    = "vault-key"

  vpc\_security\_group\_ids      = var.sg

  subnet\_id                   = var.subnet

 associate\_public\_ip\_address = "true"

  user\_data                   = file("vault.sh")

  tags = {

    Name = "vault"

  }

}

1. the IP address of an EC2 instance running in AWS
2. all possible data of a specific Amazon Machine Image(AMI) from AWS
3. the latest AMI you have previously used for an Amazon Linux 2 image
4. a custom AMI for Amazon Linux 2

**21.Likitha is calling a child module to deploy infrastructure for her organization. Just as a good architect does (and suggested by HashiCorp), she specifies the module version she wants to use even though there are newer versions available. During a terrafominit, Terraform downloads v0.0.5 just as expected.**

**What would happen if Marga removed the version parameter in the module block and ran a terraforminit again?**

module "consul" {

 source  = "hashicorp/consul/aws"

 version = "0.0.5"

 servers = 3

}

1. Terraform would return an error, as the version parameter is required
2. Terraform would skip the module
3. Terraform would use the existing module already downloaded.
4. Terraform would download the latest version of the module

**22. Based on the following code, which code block will create a resource first?**

resource "aws\_instance" "data\_processing" {

 ami           = data.aws\_ami.amazon\_linux.id

 instance\_type = "t2.micro"

 depends\_on = [aws\_s3\_bucket.customer\_data]

}

module "example\_sqs\_queue" {

 source  = "terraform-aws-modules/sqs/aws"

 version = "2.1.0"

 depends\_on = [aws\_s3\_bucket.customer\_data, aws\_instance.data\_processing]

}

resource "aws\_s3\_bucket" "customer\_data" {

 acl = "private"

}

resource "aws\_eip" "ip" {

 vpc      = true

 instance = aws\_instance.data\_processing.id

}

1. aws\_s3\_bucket.customer\_data
2. aws\_eip.ip
3. aws\_instance.data\_processing
4. Example\_sqs\_queue

**23 A user is using Terraform to deploy infrastructure using modules. Where is the module below stored?**

module "monitoring\_tools" {

 source = "./modules/monitoring\_tools"

 cluster\_hostname = module.k8s\_cluster.hostname

}

1. in a public GitLab repository
2. locally on the instance running Terraform
3. on the Terraform public registry
4. a private registry in Terraform Cloud (free)

**24.What happens if multiple users attempt to run a terraform apply simultaneously when using a remote backend? (select two)**

1. if the backend does not support locking, the state file could become corrupted
2. if the backend supports locking, the first terraform apply will lock the file for changes preventing the second user from running the apply
3. both users will get an error
4. the Terraform apply will work for both users

**25.Given a Terraformconfig that includes the following code, how would you reference the last instance that will be created?**

resource "aws\_instance" "database" {

 # ...

 for\_each = {

   "vault": "secrets",

   "terraform": "infrastructure",

   "consul": "networking",

   "nomad": "scheduler"

 }

}

1. aws\_instance.database[4]
2. aws\_instance.nomad
3. aws\_instance.database["nomad"]
4. Aws\_instance.database[2]

**26.You have a Terraform configuration file with no defined resources. However, there is a related state file for resources that were created on AWS. What happens when you run a *terraform apply*?**

1. Terraform will not perform any operations.
2. Terraform will scan the AWS infrastructure and create a new configuration file based on the state file.
3. Terraform will produce an error since there are no resources define
4. Terraform will destroy all of the resources

**27. When a terraform apply is executed, where is the AWS provider retrieving credentials to create cloud resources in the code snippet below?**

provider "aws" {

region = us-east-1

access\_key = data.vault\_aws\_access\_credentials.creds.access\_key

secret\_key = data.vault\_aws\_access\_credentials.creds.secret\_key

}

1. from the .tfvars file called vault
2. from a data source that is retrieving credentials from HashiCorp Vault. Vault is dynamically generating the credentials on Terraform's behalf.
3. from a variable called vault\_aws\_access\_credentials
4. from a script that is executing commands against Vault

**28.There are multiple ways to provide sensitive values when using Terraform. However, sensitive information provided in your configuration can be written to the state file, which is not desirable. Which method below will not result in sensitive information being written to the state file?**

1. using a declared variable
2. using a tfvars file
3. retrieving the credentials from a data source, such as HashiCorp Vault
4. none of the above

**29.Based on the Terraform code below, what block type is used to define the VPC?**

* vpc\_id = aws\_vpc.main.id

...

1. data block
2. locals block
3. resource block
4. provider block

**30. Given the following snippet of code, what will the value of the "Name" tag equal after a terraform apply?**

variable "name" {

description = "The username assigned to the infrastructure"

default = "data\_processing"

}

variable "team" {

description = "The team responsible for the infrastructure"

default = "AIS Team"

}

locals {

name = (var.name != "" ? var.name :random\_id.id.hex)

owner = var.team

common\_tags = {

Owner = local.owner

Name = local.name

}

}

1. a random hex value
2. an empty string
3. data\_processing
4. AIS Team

**31.What function does the terraform init -upgrade command perform?**

1. upgrades the backend to the latest supported version
2. upgrades all of the referenced modules and providers to the latest version of Terraform
3. update all previously installed plugins and modules to the newest version that complies with the configuration’s version constraints
4. upgrades the Terraform configuration file(s) to use the referenced Terraform version

**32.True or False? A remote backend configuration is required for using Terraform.**

1. False
2. True

**33. You are working with a cloud provider to deploy resources using Terraform. You've added the following data block to your configuration. When the data block is used, what data will be returned?**

data "aws\_ami" "amzlinux2" {

  most\_recent = true

  owners      = ["amazon"]

  filter {

    name   = "name"

    values = ["amzn21-ami-hvmm-\*-x86\_64-ebs"]

  }

}

resource "aws\_instance" "vault" {

ami                         = data.aws\_ami.amzlinux2.id

instance\_type               = "t3.micro"

  key\_name                    = "vault-key"

  vpc\_security\_group\_ids      = var.sg

  subnet\_id                   = var.subnet

  associate\_public\_ip\_address = "true"

  user\_data\_file                   = file("vault.sh")

  tags = {

    Name = "vault"

  }

}

1. all possible data of a specific Amazon Machine Image(AMI) from AWS
2. the latest AMI you have previously used for an Amazon Linux 2 image
3. the IP address of an EC2 instance running in AWS
4. a custom AMI for Amazon Linux 2

**34.Which of the following is not a benefit of Terraform state?**

1. reduces the amount of outbound traffic by requiring that state is stored locally
2. determines the dependency order for deployed resources
3. increases performance by reducing the requirement to query multiple resources at once
4. provides a one-to-one mapping of the configuration to real-world resources

**35. There are an endless number of benefits to using Terraform within your organization. Which of the following are true statements regarding Terraform? (select three)**

1. Terraform is cloud-agnostic but requires a specific provider for the cloud platform
2. Terraform can manage dependencies within a single cloud, but not cross-cloud
3. A single Terraform configuration file can be used to manage multiple providers
4. Terraform can simplify both management and orchestration of deploying large-scale, multi-cloud infrastructure

**36. After using Terraform locally to deploy cloud resources, you have decided to move your state file to an Amazon S3 remote backend. You configure Terraform with the proper configuration as shown below. What command should be run in order to complete the state migration while copying the existing state to the new backend?**

terraform {

backend "s3" {

bucket = "tf-bucket"

key = "terraform/krausen/"

region = "us-east-1"

}

}

1. terraform apply -refresh-only
2. terraforminit -migrate-state
3. terraform plan –replace
4. terraform state show

**37. Your team is using Terraform, and multiple team members need to be able to manage the infrastructure. You need to support state locking to reduce the chance of corrupting the state file. What backends can you use to meet these requirements? (select three)**

1. kubernetes backend
2. s3 backend (with DynamoDB)
3. consul backend
4. local backend

**38. I want to create a sample file on an EC2 instance using Terraform. How can I connect Terraform to the EC2 instance?**

1. by using local

2. Lifecycle block

3. connect provisioners

4. State configurations

**39. You are working on updating your infrastructure managed by Terraform. Before lunch, you update your configuration file and run a terraform plan to validate the changes. While you are away, a colleague manually updates a tag on a managed resource directly in the console (UI).**

**What will happen when you run a terraform apply?**

1. Before applying the new configuration, Terraform will refresh the state and recognize the manual change. It will update the resource based on the desired state as configured in the Terraform configuration. The manual change will no longer exist.
2. Terraform will recognize the manual change and return an error since the Terraform state no longer matches the real-world infrastructure.
3. Terraform will destroy the manually-changed resource and recreate it to ensure the infrastructure matches the desired state.
4. Terraform will update the manually changed resource back to the original configuration. It will then apply the new changes defined in the updated configuration file.

**40 .Let’s say you have created an RDS  usingTerraform and someone does the manual change on it next time you run Terraform plan. What will happen?**

1. Manual changes will be gone
2. Changes will be updated in config file
3. No changes
4. Existing changes will be deleted and recreated again.

**41. If I have created a s3 bucket and aws lambda  usingTerraform and unfortunately the tfstate file got deleted, can you recover it? If yes which option is helpful (State File is only on the local machine not on s3)**

1. Terraform data source
2. Terraform Import
3. We can’t retrieve
4. Create resources from scratch.

**42.InTerraform, what is the purpose of the depends\_on attribute within a resource block?**

a) To specify dependencies between resources.

b) To define conditional logic for resource creation.

c) To enable parallel execution of resource provisioning.

d) To enforce resource deletion order during destruction.

**43. User is creating AWS Ec2 instance he wants to display information about resources created by Terraform after execution**

a. Terraform Dependent block

b. Terraform outputs

c. Terraform data source

d. Terraform Import

**44. User want to create in same stack two s3 buckets one is in ap-south-1 and one is in us-east-1 give the possibility.**

a. Two config files with different regions

b. Alias providers

c. Dependent block

d. Two times need to run config files.

**45. user want to avoid accidental / unwanted recreation of resource instances when a input list (or just it’s order) has been modified. So which parameter is help full here.**

**Note:** user want to create three s3 buckets so to avoid the above issues which is the good practice

a. Count

b. for each

c. both

d. dynamo locking

**46. What does the terraform validate command do?**

a) It checks the syntax and configuration of Terraform files.

b) It applies the changes described in a Terraform configuration.

c) It refreshes the state

d) validate the remote state file.

**47. A user tried to destroy the Terraform configuration file using the terraform destroy command, but it is not working. What could be the reason?**

a) lifecycle enabled

b) Terraform init missing before

c) Local block enabled

d) No resource Iin config file

**48. Two developers working with same project repo so one developer started terraform apply command (state file- backend and DynamoDB lock configured) and other developer given terraform validate so what would be the expected output for developer-2 end.**

1. lock acquired

2. Try again since process is going on

3. Terraform validate work without any issue

4. developer-2 need to wait till complete the developer-1 applied request.

**49. What does the terraform plugin will do?**

a. The plugin is responsible for converting the HCL code into API calls and sends the request to the appropriate provider (ex:AWS, GCP, Azure)

b. The plugin is responsible for communicate to terraform registry

c. The plugin is responsible for converting the the appropriate provider (ex:AWS, GCP, Azure) into HCl code

d. both a and C

**50. What does the Remote exec provisioner will do?**

a. Run commands using Terraform on a remote server

b. Run commands using Terraform on the local system

c. Run commands using both local and remote

d. None of the above

**51. What is the use of Terraform module**

a To make to code standardized

b To reduce the code duplication

c We can create the terraform modules one time and reuse them whenever needed

d all of the above

**52.How I can delete/destroy specific resources without changing logic**

a. We need to taint that particular resource to be deleted using terraform taint RESOURCE\_TYPE.RESOURCE\_NAME command

b. deletes manually reapply the configuration file

c. Not possible to delete particular resource

d. use the work space to delete

**53. How to maintain different state files in local repo**

a. by using terraform refresh

b. by using workspace

c. Not possible only one state file

d. creates multiple repositories inside

**54. developer-1 created an EC2 instance using Terraform and developer-2 does the manual change on it next time you run Terraform plan what will happen?**

**a.** Terraform state will be mismatched and terraform will modify the EC2 instance to the desired state i.e. whatever we have defined in the .tf file

b. Delete all recreate again

c. Terraform state nothing to do no changes

d, Terraform state will be track and terraform will modify the EC2 instance to the desired state i.e. whatever there in remote infra

**55.What is terraform state locking will do?**

**a.** Whenever we are working on any terraform code and do terraform plan, apply or destroy terraform will lock the state file in order to prevent the destructive action.

b. Whenever we are working on any terraform code and do terraforminit, validate so terraform will lock the state file in order to prevent the destructive action.

c. Both a, b

d. Whenever we are working on any terraform code and do terraform delete, refresh so terraform will lock the state file in order to prevent the destructive action.

**56. which is the best practice to store state file**

a. S3 storage if it is aws

b. Github

c. a,b

d. local

**57. Explain about Terraform null resource and perform any sample lab.**

-------------------------All The Best ----------------------------