**Terraform:**

**1. terraform init initializes a sample main.tf file in the current directory.**

1. True
2. False

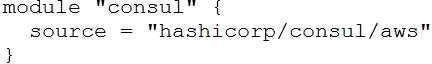
**2. Why would you use the terraform taint command?**

1. When you want to force Terraform to destroy a resource on the next apply
2. When you want to force Terraform to destroy and recreate a resource on the next apply
3. When you want Terraform to ignore a resource on the next apply
4. When you want Terraform to destroy all the infrastructure in your workspace

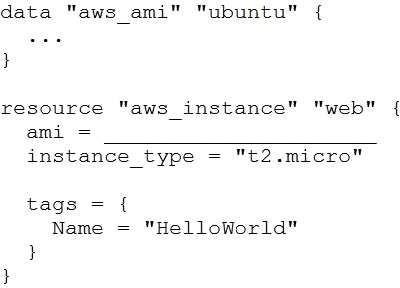
**3. Which of the following is the correct way to pass the value in the variable num\_servers into a module with the input servers?**

1. servers = num\_servers
2. servers = variable.num\_servers
3. servers = var(num\_servers)
4. servers = var.num\_servers

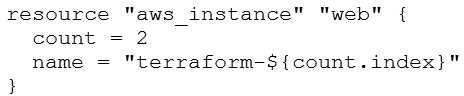
**4. How do you specify version 1.0.0?**



1. Modules stored on the public Terraform Module Registry do not support versioning
2. Append ?ref=v1.0.0 argument to the source path
3. Add version = "1.0.0" attribute to module block
4. Nothing – modules stored on the public Terraform Module Registry always default to version 1.0.0

**5. Examine the following Terraform configuration, which uses the data source for an AWS AMI. What value should you enter for the ami argument in the AWS instance resource?**  


1. aws\_ami.ubuntu
2. data.aws\_ami.ubuntu
3. data.aws\_ami.ubuntu.id
4. aws\_ami.ubuntu.id

**6. How would you reference the "name" value of the second instance of this fictitious resource?**  


1. element(aws\_instance.web, 2)
2. aws\_instance.web[1].name
3. aws\_instance.web[1]
4. aws\_instance.web[2].name
5. aws\_instance.web.\*.name

**7. What is terraform refresh intended to detect?**

1. Terraform configuration code changes
2. Empty state files
3. State file drift
4. Corrupt state files

**8. Which option can not be used to keep secrets out of Terraform configuration files?**

1. A Terraform provider
2. Environment variables
3. A -var flag
4. secure string

**9. You have deployed a new webapp with a public IP address on a clod provider. However, you did not create any outputs for your code.  
What is the best method to quickly find the IP address of the resource you deployed?**

1. Run terraform output ip\_address to view the result
2. In a new folder, use the terraform\_remote\_state data source to load in the state file, then write an output for each resource that you find the state file
3. Run terraform state list to find the name of the resource, then terraform state show to find the attributes including public IP address
4. Run terraform destroy then terraform apply and look for the IP address in stdout

**10. What is the workflow for deploying new infrastructure with Terraform?**

1. terraform plan to import the current infrastructure to the state file, make code changes, and terraform apply to update the infrastructure
2. Write a Terraform configuration, run terraform show to view proposed changes, and terraform apply to create new infrastructure.
3. terraform plan to import the current infrastructure to the state file, make code changes, and terraform apply to update the infrastructure
4. Write a Terraform configuration, run terraform init, run terraform plan to view planned infrastructure changes, and terraform apply to create new infrastructure.

**11. Which two steps are required to provision new infrastructure in the Terraform workflow? (Choose two.)**

1. Destroy
2. Apply
3. Import
4. Init
5. Validate

**12. The terraform plan command applies the configuration changes to your infrastructure.**

A. True

B. False

**13. What does the terraform init command do?**

A. Applies the configuration changes

B. Initializes a working directory containing Terraform configuration files

C. Updates the state file

D. Deletes unused resources

**14. Which file contains sensitive outputs and resource state in Terraform?**

A. terraform.tfstate

B. main.tf

C. variables.tf

D. outputs.tf

**15. What is the default file extension for Terraform configuration files?**

A. .yaml

B. .hcl

C. .tf

D. .json

**16. True/False: Terraform always requires internet access to apply a configuration (****e.g localstak).**

A. True

B. False

**17. What is the purpose of the terraform lock file (.terraform.lock.hcl)?**

A. To store state information

B. To record the exact versions of providers used for configuration

C. To enforce security policies

D. To manage module dependencies

**18. Which of the following is NOT a valid Terraform resource lifecycle argument?**

A. create\_before\_destroy

B. prevent\_destroy

C. depends\_on

D. ignore\_changes

**19. How can you fetch outputs from a remote Terraform state?**

A. Use the terraform import command

B. Define a terraform\_remote\_state data source

C. Manually copy the state file

D. Use the terraform fetch command

**20. Which of these options enables version control for a Terraform module?**

A. Store the module in a Terraform workspace

B. Use Git tags or branches in the module source URL

C. Use the version block in the configuration

D. Download modules manually

**21. Which type of backend enables collaboration on Terraform state?**

A. Local

B. Remote

C. GitHub

D. FTP

**22.How does Terraform handle state drift?**

A. Automatically updates the configuration file to match the drift

B. Marks the resource as tainted

C. Requires a manual refresh or plan to detect drift

D. Re-applies the configuration to resolve the drift

**23. What happens if you delete the terraform.tfstate file and re-run terraform apply?**

A. Terraform will create new resources to match the configuration

B. Terraform will recreate the state file with existing resources

C. Terraform will throw an error and refuse to apply changes

D. Terraform will use the backup file to restore the state

**24. How can you customize the execution of specific providers in Terraform?**

A. By using the terraform-provider command

B. Specifying the provider\_alias argument in the provider block

C. Using meta-arguments like custom\_config

D. It is not possible to customize provider execution

**25. How would you share variables between multiple modules in a configuration?**

A. Use global variables declared in the root module

B. Export outputs from one module and pass them as inputs to another

C. Store variable values in the backend state file

D. Use a shared state data source

**26. What is the purpose of the depends\_on meta-argument in Terraform?**

A. To force Terraform to destroy dependent resources first

B. To explicitly specify dependencies between resources

C. To manage module dependencies

D. To override lifecycle settings

**27. What does the for\_each argument allow you to do in Terraform?**

A. Loop through a list or map to create multiple resource instances

B. Reference attributes of another resource

C. Filter resources in a state file

D. Apply configuration to a subset of resources

**28. In which scenario would you use the terraform import command?**

A. To load remote state files

B. To import existing infrastructure into Terraform management

C. To migrate resources between workspaces

D. To refresh the state file

**29. Which attribute ensures a resource is recreated during every terraform apply?**

A. replace\_on\_update

B. force\_recreate

C. always\_recreate

D. None of the above

**30. How do Terraform modules promote reusability?**

A. By enforcing versioning on modules

B. By allowing variables and outputs to be parameterized

C. By creating isolated execution environments

D. By storing state files in separate locations

**31. What is a potential risk when using multiple Terraform workspaces?**

A. Resource name conflicts

B. Inconsistent state management across workspaces

C. Lack of backend support for workspaces

D. Workspaces cannot store sensitive data

**32. Which command can repair a corrupted state file?**

A. terraform validate

B. terraform repair

C. terraform force-unlock

D. terraform state push

**33. What is the purpose of Terraform dynamic blocks?**

A. Automate resource creation based on user input

B. Generate nested blocks in a resource or module dynamically

C. Allow runtime variable declarations

D. Store runtime state information