===>>>Explain core terraform end to end workflow to deploy and delete resource in Azure or any cloud provider?

1) we write a code as per our requirement

2) TF init command will initialize working directory containg terraform configuration files, dot .terrraform file will be created in that files provider files will be there and required plugins will be added.

3) TF validate will check systax of your code

4) TF fmt will make beautifully alighn the code

4) TF plan will create the resources, its like a preview before applying, what is going to be created

5) TF apply --auto-approve will create your resource and terraform.tfstate file is created where all resources are update and create.

===>>>Explain or walk through step process to secure .tfstate file and by making it readily avaiable for other developers within the team?

First we need to create a backend block , we will mention where our Tfstatefile will be store eg in stroage account inside container. ITs securely store, helps to keep version control and helps in locking mechnaism. whoecer first apply he will aquire lease means nobody can acces statefile. To unlock we have break lease or through cli tf force -unlock id here (For aws, dynamo DB is stored a data has a key value pair in table format means db is lock, Now second dev cannot have access to statefile so there will be no conflict)

Q. Do you know state file in Terraform?

A. Yes, I'm familiar with the state file in Terraform. The state file is a important component of Terraform that keeps track current state of our infrastructure as managed by Terraform. It includes information about resources, their attributes, dependencies, and other details.

Q. Where are you store the state file in your project?

A. In our project, we store Terraform state file remotely, typically we use a cloud storage service to store state file. This ensures that the state file is accessible and manageable by the entire team, especially when working collaboratively.

Q. What is the purpose of storing the state file remotely?

A. There are two main purposes to store state file remotely that are: collaboration and safety.

1. Collaboration: Ex: - Suppose multiple people are working on the same project. If we store state file remotely so everyone of team member can access and update it. This ensures that everyone is on the same page and prevents conflicts when making changes to the infrastructure.

2. Safety: Ex: - Let’s say, if anyone accidentally delete the local copy of the state file. If it's stored remotely, we can recover it easily. Also, by using a cloud service like Azure Storage, I have built-in versioning and backup features, adding an extra layer of safety.

Q. Concept of locking in Terraform? OR What is the purpose of state lockfile?

A. Locking in Terraform is a mechanism to ensure that only one person can modify the project/code at a time, preventing conflicts or confusion. Once the changes are applied, the lock is removed, and your friend can then make their changes with their own lock.

Q. Can you tell me the purpose of the data block in Terraform?

A. The purpose of Data block in Terraform is used to retrieve information from various sources, such as cloud providers.

For Ex- If we have already created resource in azure portal and want to use in our code we can use data block to fetching information about existing resources.

===>>> Explain the various type of META-argument in terrafomr and their benefits?

Meta arguments changes the defaults behaviours of TF configuartion.

- depends\_on, count, for\_each, provider, lifecycle

-depends\_on will run your code parallely means first we need to create RG than Storage ac.

-for\_each Map will help us in looping the resources multiple time.

-provider is which platform cloud you want to use.

-lifecycle is a nested block that can appear within a Resource block eg create\_before\_destroy = true..it will recreated first than destroy

===>>> Who creates the "terraform.tfstate.backup" file amd under which scenario it is created?

-Backup state file is automatically created, when tf destroy command is executed. It will helps us to re-stroe the infrasturcture prior running a tf destroy command

===>>> Resouurce "Azure Storage account" was previouslty created manually, now the resource needs to be brought under terraform to manage. How do we achieve the same?

-We can achieve this by TF import commad. it will bring your existing infrastructure under its management by importing into terraform

- First you write terraform conf file to match the resourse created manually than run terraform import rg, sa etc

-First tf init

-second systax: terraform import terraform\_block\_lables resources\_id

-tf plan ///it will show no chnages infr matches to the conf

===>>> Resources "Azure App Service" was created and managed by TF, but someone added a few tags or made a minor changes manually on the portal. What will happpne in such cases? How tf will react?

In such cases terraform will try and delete the tags or minor changes manually because Code is true in our configuration . We just need to run tf plan and it remove changes made manually and tf refresh it will update our statefile

===>>> Recently migrated state file to Azure Stroage {blob}, now post, if i make any terraform configuration chnage I dont see any .tfstate-backup file getting created locally or remotely. which use to happen previously?

SO we have state files backed up while storing in Azure blob versioning need to be enabled at storage account level, under data management->data protection option call tracking enable versioning.

===>>> Assume terraform configuration files are deleted and I am just left with an updated .tfstate file. how do we recover the configuration using a .tfstate file, if possible?

We have multiple ways to handle the ablove scenario:

1) Analyse the .statefile and re-create configuarion from scrath OR follow the terraform import workflow and get the resources under terraform management

===>>> what is terraform workspacr and how it is beneficaila in day in day out jobs?

Workspace containd everything that TF needs to manage a given collection of infrastucture and separate workspace functions like separete working directories

-