GSS Terraform Quiz 1

1. A resource I’m looking to create has a parameter that seems like we might want to adjust in the future, however, for now, the value being used is always the same. What would be a good approach to minimize the need to configure this parameter each time we create the resource, but also enable easy adjustment in the future if we decide to change the parameter?

* Lets take a resource parameter I would like to change often for Ec2 would be instance type or AMI which we would like to change etc. Since terraform is declarative language we can declare a “TF\_VAR\_ Ec2instance\_type “ variable and we can change the instance type at that single point . or as our code is already in the form of modules we can change directly in the tfvars if the variable is declared over there , if not we can declare one .
* I see for Blitzen project we have a variable by the name :: “ecs\_instance\_type”.

1. If a secret (password, ssh key, etc) is needed to create a resource in an environment, what is the correct approach to create and use this secret within Terraform?

* We will use the secrets stored in the parameter store, if required a new one we will be creating one new parameter with standard naming criteria as we have defined in the code in the file loadsecret.py in binaries of Enviroment folder .
* We will be calling this secrets using variables “TF\_VAR\_project” and “TF\_VAR\_project\_type” from init.vars folder which inturn will be used in init.sh script.
* This way helps us in not exposing our secrets to public when we host our code in Git repos and even state files will not be saving I believe I am not sure but I need to read more about how state files work and what these will store.

1. I’m setting up a new application. Some resources needed by the application only need to be created once, while others need to have a resource created each time we stand up a new environment. What are some options we have available to allow for the correct resources to be created?

* I see this as something related to setting up of GC Client environments where we need to create loadbalancers for once but we need to create targetgoups for each client . I am not clear if I have understood this correctly but I am still working on this .

1. Give an example of the steps needed to run the command terraform plan after opening a GSS Terraform project in VSCode.

* First check the versions.tf to make sure that we are running terraform version higher than 0.12.X. for safer side always pull the project from git to get latest code changes done by others .
* Navigate to base folder in the environment folder and run the command ::

.init.sh {environment name either dev or prod or qa}

* Check if the entire project is downloaded in to the terminal for compiling .
* If everthing looks good with out any error, they run the command ::terraform apply
* This gives the entire plan of execution of terraform and changes or creations or destructions of any resources of our AWS environment . if all are as per our expectations , enter YES at the bottom as input and this will apply all changes . check if execution throws any error and modify accordingly .