Terraform provisiners

- Terraform provides 3 kinds of provisioners
  - local-exec: used to execute some script on local machine where terraform is executing
  - remote-exec: used to execute some script on remote machine
  - file: this is used to copy files from local to remote
- Refer Here for official docs
- When connecting to remote machines we need to establish connections

Activity

- Create an ec2 instance and execute script using provisioners
- Provisioners will also be executed only once when the resource is created.
- Refer Here for changes

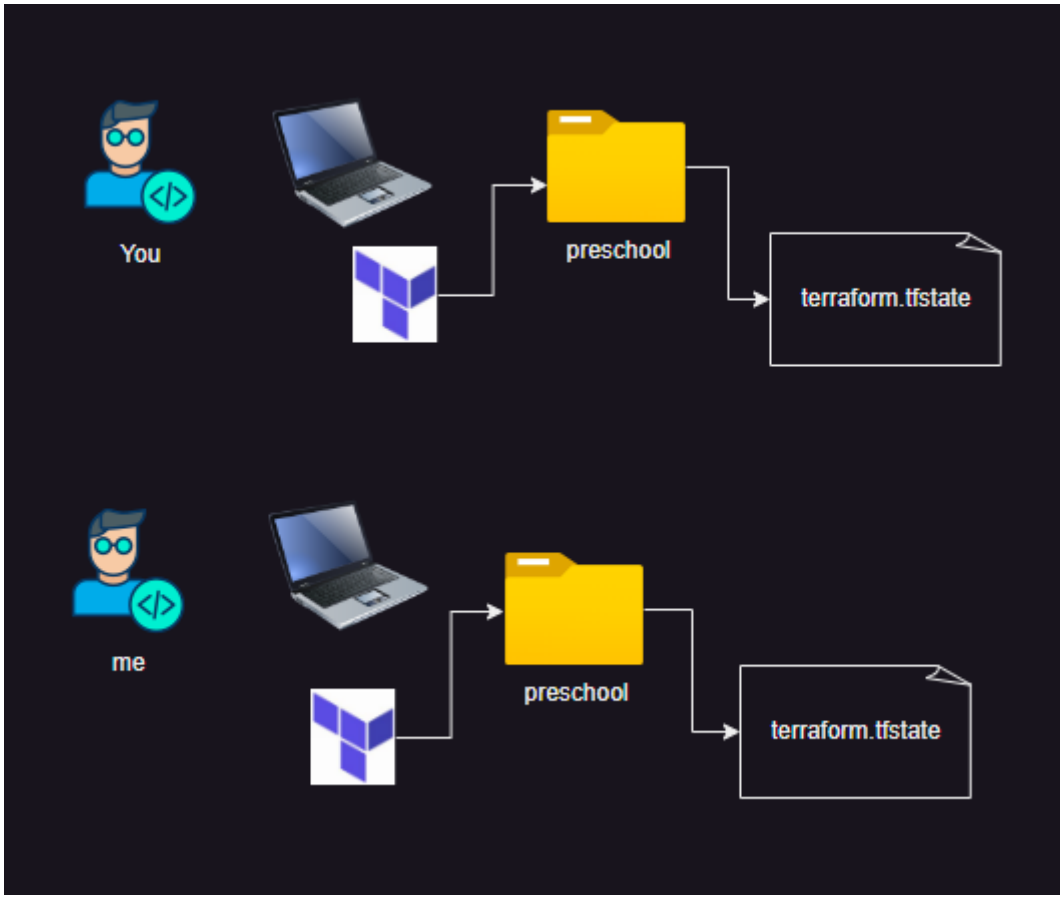
Activity Improvement: I want to execute the install script without recreating ec2 instance.

- For acheiving terraform has an interesting resource called as `null_resource`
- We will move provisioning into null resource and trigger the script execution whenever we want. Refer Here
- Refer Here for changes

Terraform Backends

- Backend is a location where terraform state is stored. The default backend is local folder

- Consider the following scenario



- Since the state is stored locally, two different executions will lead to terraform trying to create 2 different set of resources
- Solution to this problem is to have common location for both engieers to store the state file
- Refer Here for official docs
- There are different backends available
  - local
  - remote
  - s3
  - azurerm
  - ...
- s3 backend doesnot support locking, for locking in s3 we need to create a dynamob db

Configuring s3 backend

- Create an s3 bucket with some unique name `qtterraformbackend`
- Now create a dynamodb table with partitionKey `LockID`
- Refer Here for the changes done