# Option-1 (Always provide -var for both plan and apply)

# Review the terraform plan

terraform plan -var="ec2\_instance\_type=t3.large" -var="ec2\_instance\_count=1"

# Create Resources (optional)

terraform apply -var="ec2\_instance\_type=t3.large" -var="ec2\_instance\_count=1"

A screenshot of a computer program

Description automatically generated

In the code I mentioned default for instance count to be 2, and type to be t3.micro.

But when I applied terraform plan I mentioned

terraform plan -var="ec2\_instance\_type=t3.large" -var="ec2\_instance\_count=1"



But as 1 count was mentioned and type is mentioned t3.large so the configuration has been changed and only 1 count will be executed for t3.large.

# Option-2 (Generate plan file with -var and use that with apply)

# Generate Terraform plan file

terraform plan -var="ec2\_instance\_type=t3.large" -var="ec2\_instance\_count=1" -out v3out.plan

# Create / Deploy Terraform Resources using Plan file

terraform apply v3out.plan

lets say in terraform variables, default for count you have set the value to ‘2’.

So you can use -var=”ec2\_instance\_count=1” when you apply terraform plan

In option 2, you make a separate plan file in order to assign the variables and then execute the file giving new configurations to your instances.

A screenshot of a computer

Description automatically generated

Here the plan file was created with the configuration that we set in the CLI.

A screenshot of a computer program

Description automatically generated

While you make this plan file you wont need to write the same configuration when you apply. Simply terraform apply “file name” can be executed. In this case

terraform apply v3out.plan.