# Introduction

https://www.youtube.com/watch?v=0zo45jvYQW0&list=PLOa-edppsqFm10V0vh-szO7YffAKcW5K-&index=3

Terraform provides infrastructure as code facility

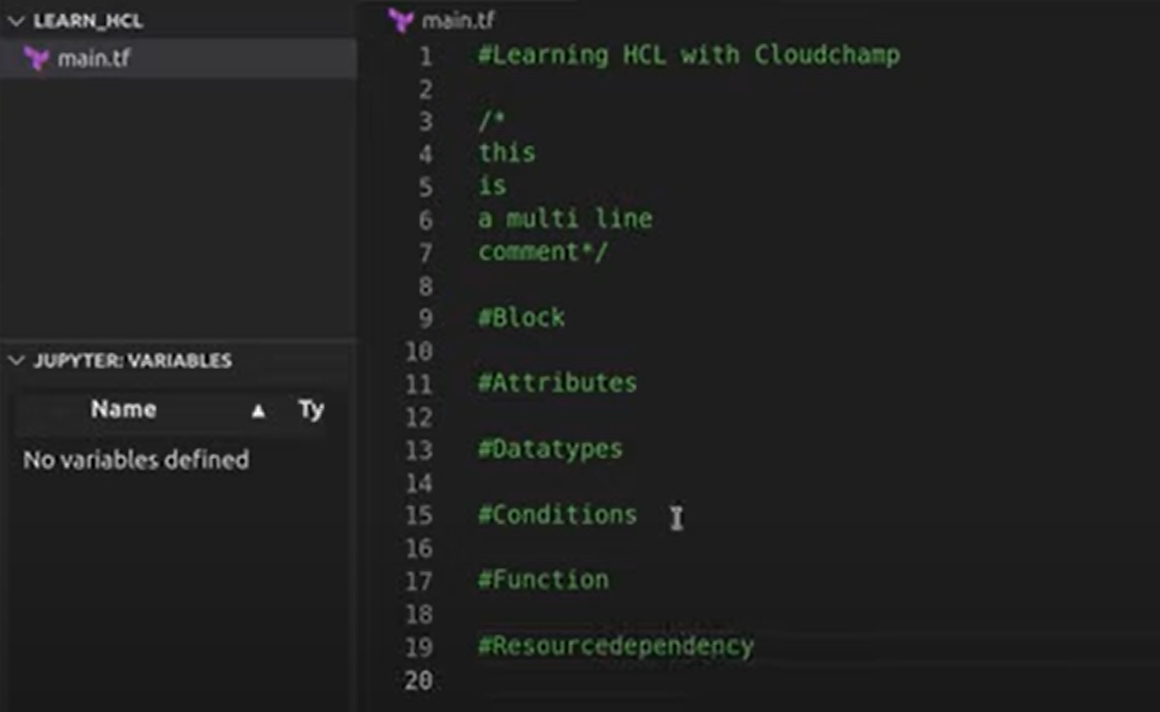
Terraform have two component core , providers

CORE :- config file and state

PROVIDER:- aws etc.

Hashicorp Configuration language HCL is used to write tf config files

We will learn below points like blocks ,attributes ,dataTypes and so on



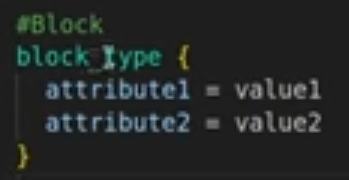
1. Block :- block is contain for other content



Resource block have some arguments then it starts with braces

There are many forms of block like provider,resource,variable,output,data,locals block in terraform

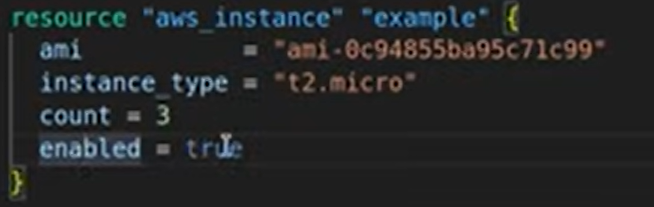
Simple syntax of block



Parameters are optional after block you might have braces directly

1. Attributes:- these are key value pairs

Lets understand block and attribute by example of creating an instance



Both ami and instance\_type is attribute here

1. Data types

Above we have string in double quotes

Boolean ,number, list,maps

List=[“items1”,”items2”]

Below we have map in a variable block

A black background with white text

Description automatically generated

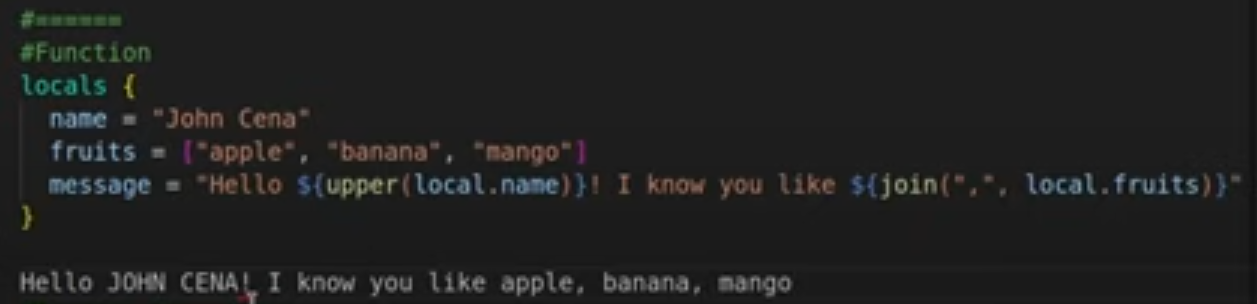
A black background with text

Description automatically generated

1. Conditions



1. Functions:-

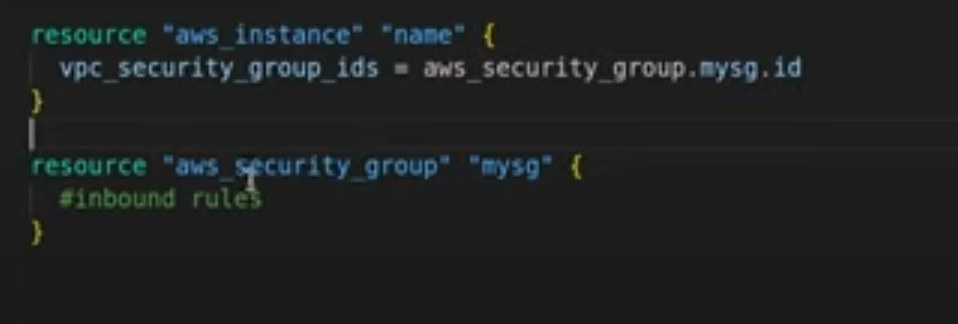


1. Resource Dependency: - represent relationship between two different resources

Two type

1) implicit :- done by terraform itself

Implicit Example instance here depends on sg



2) explicit :- depends on meta argument is used to define this to define order of dependencies

Commands

Terraform init

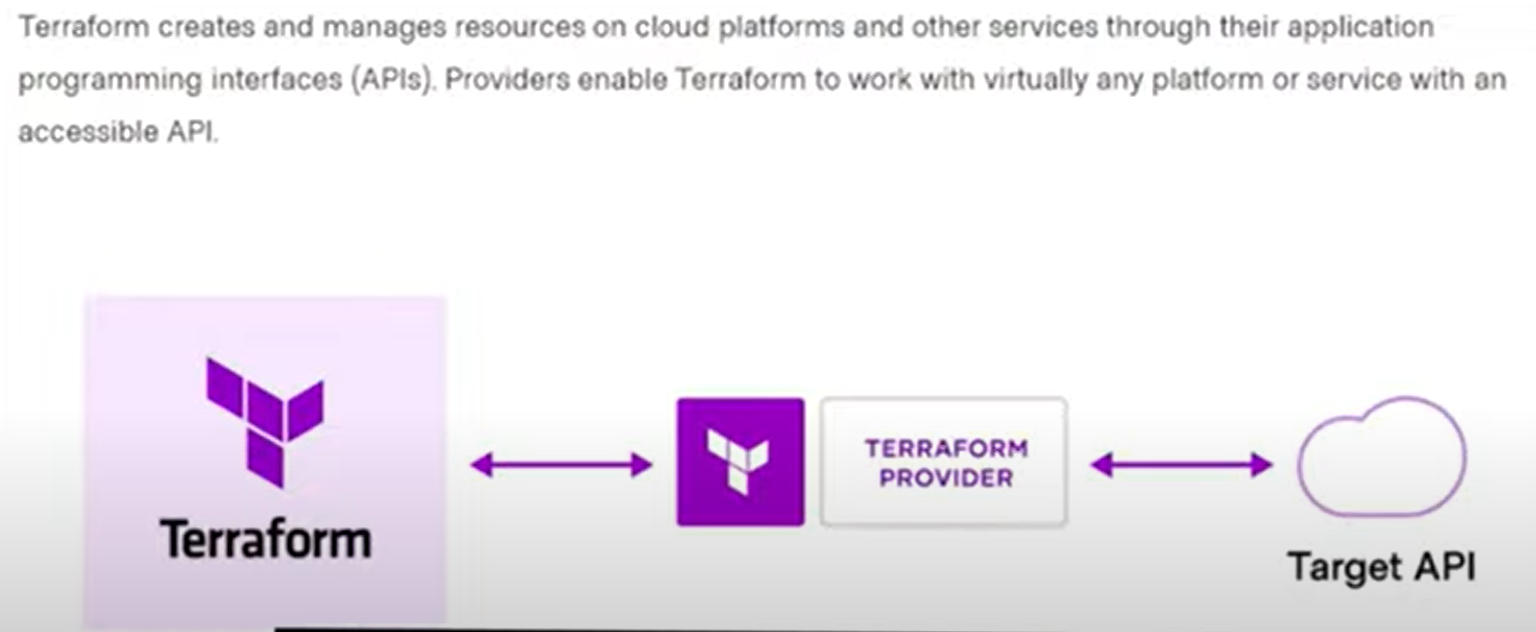
Terraform plan

Terraform apply

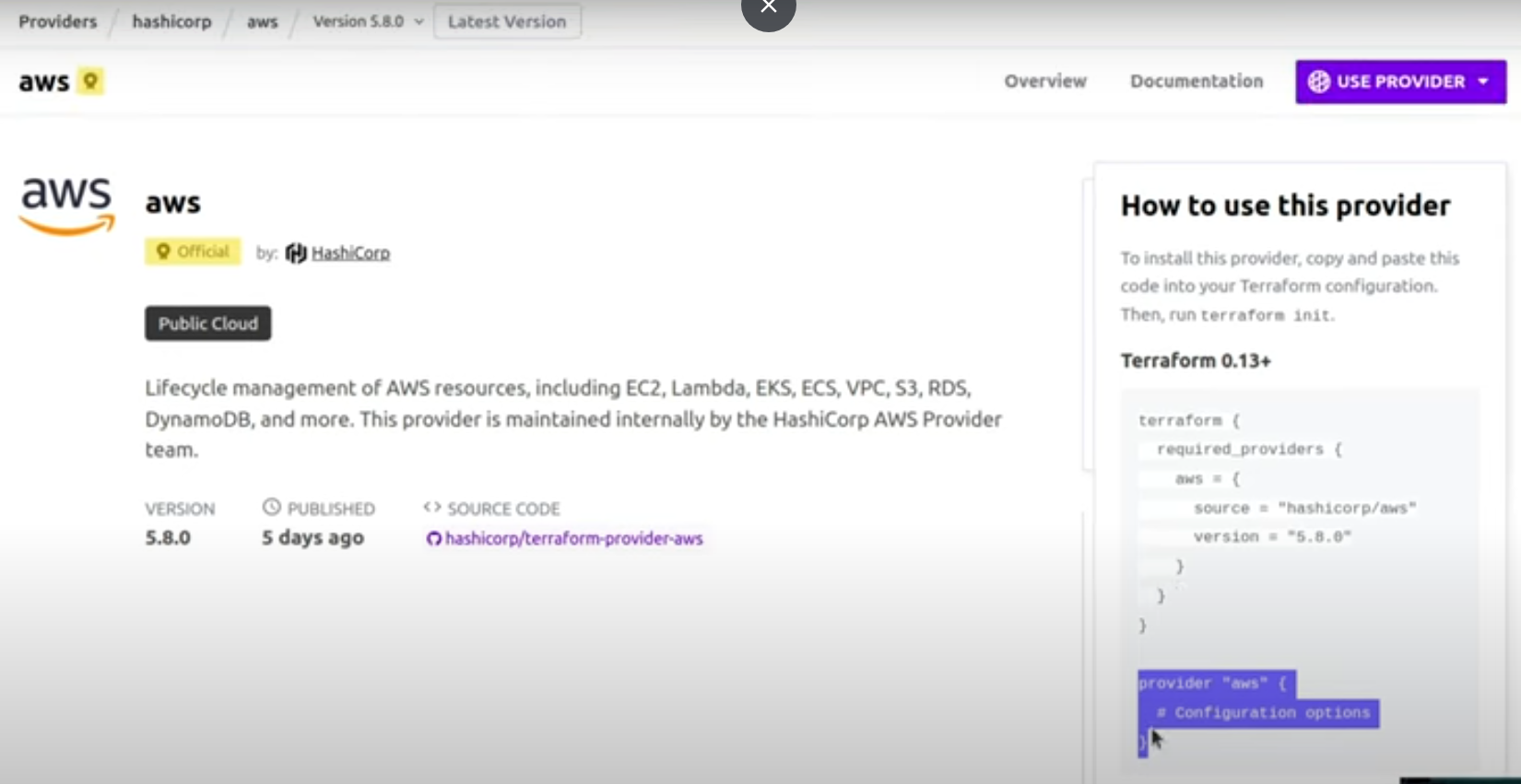
Terraform apply -auto-approve

# Providers

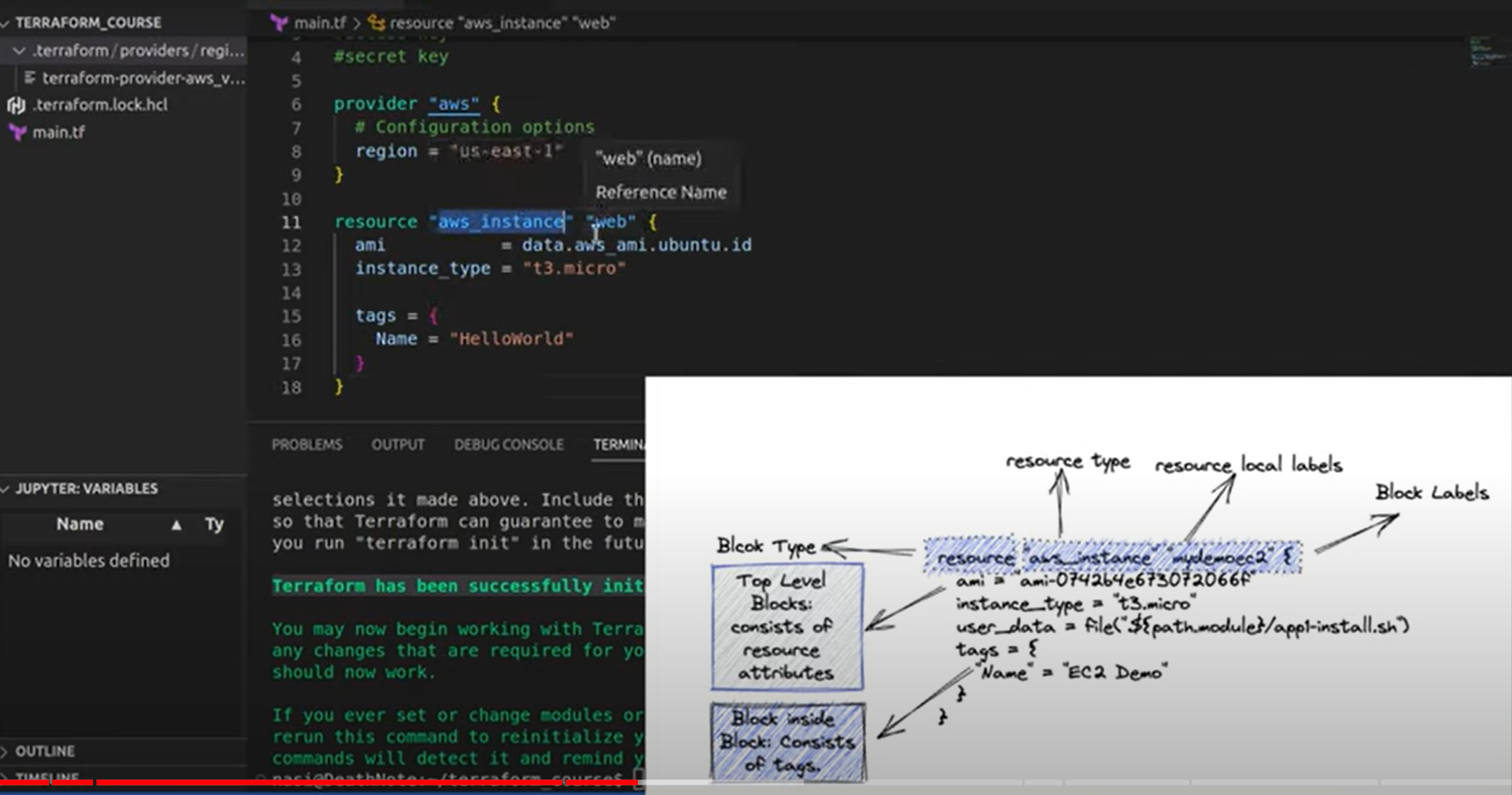
These are the first thing you define when creating resources in terraform z



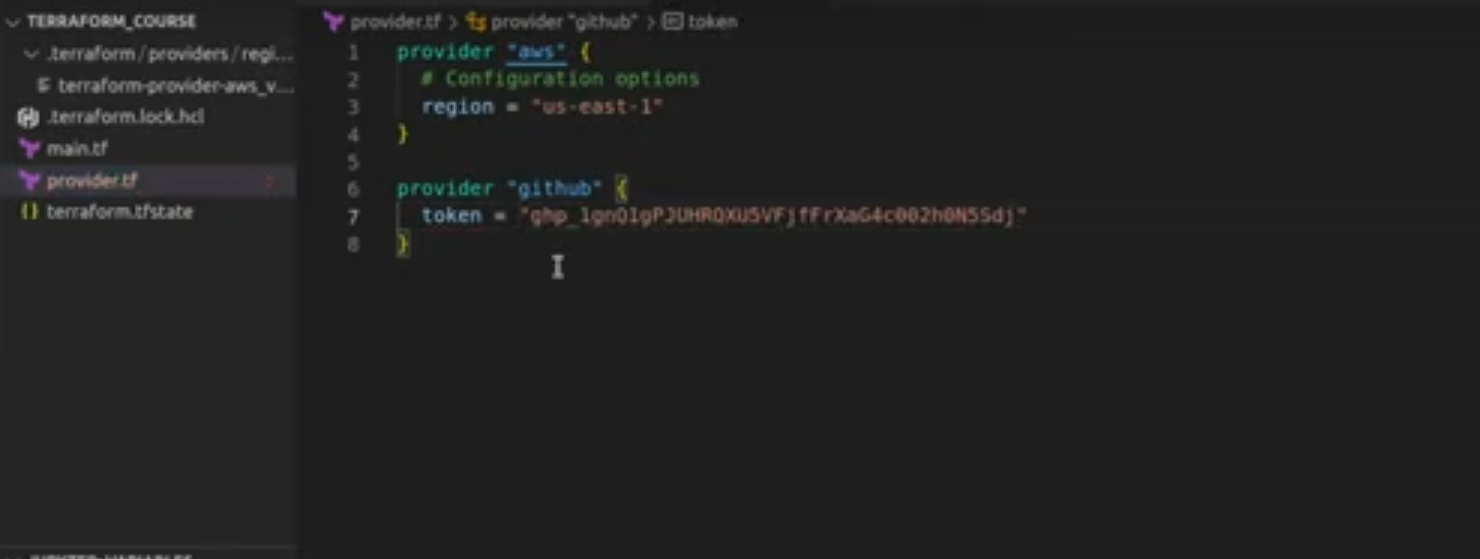
Pick a provider and put this below code in main.tf to define provider



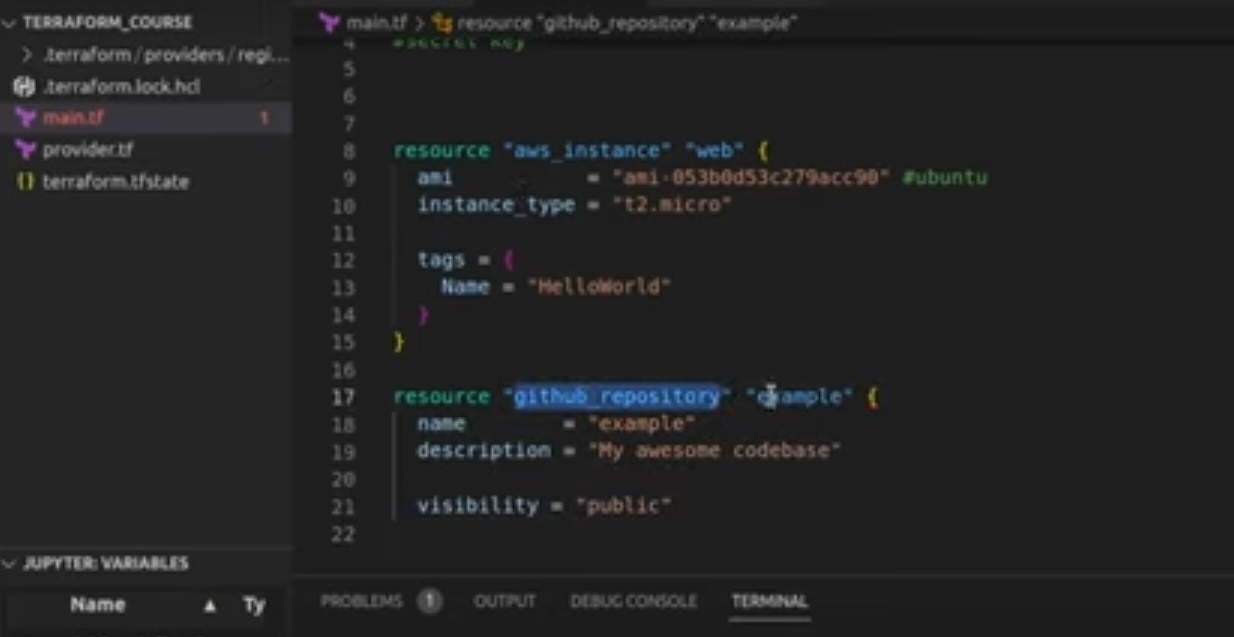
You should provide aws credentials also along with it either from aws config or for directly in provider block



We have moved all providers in one file



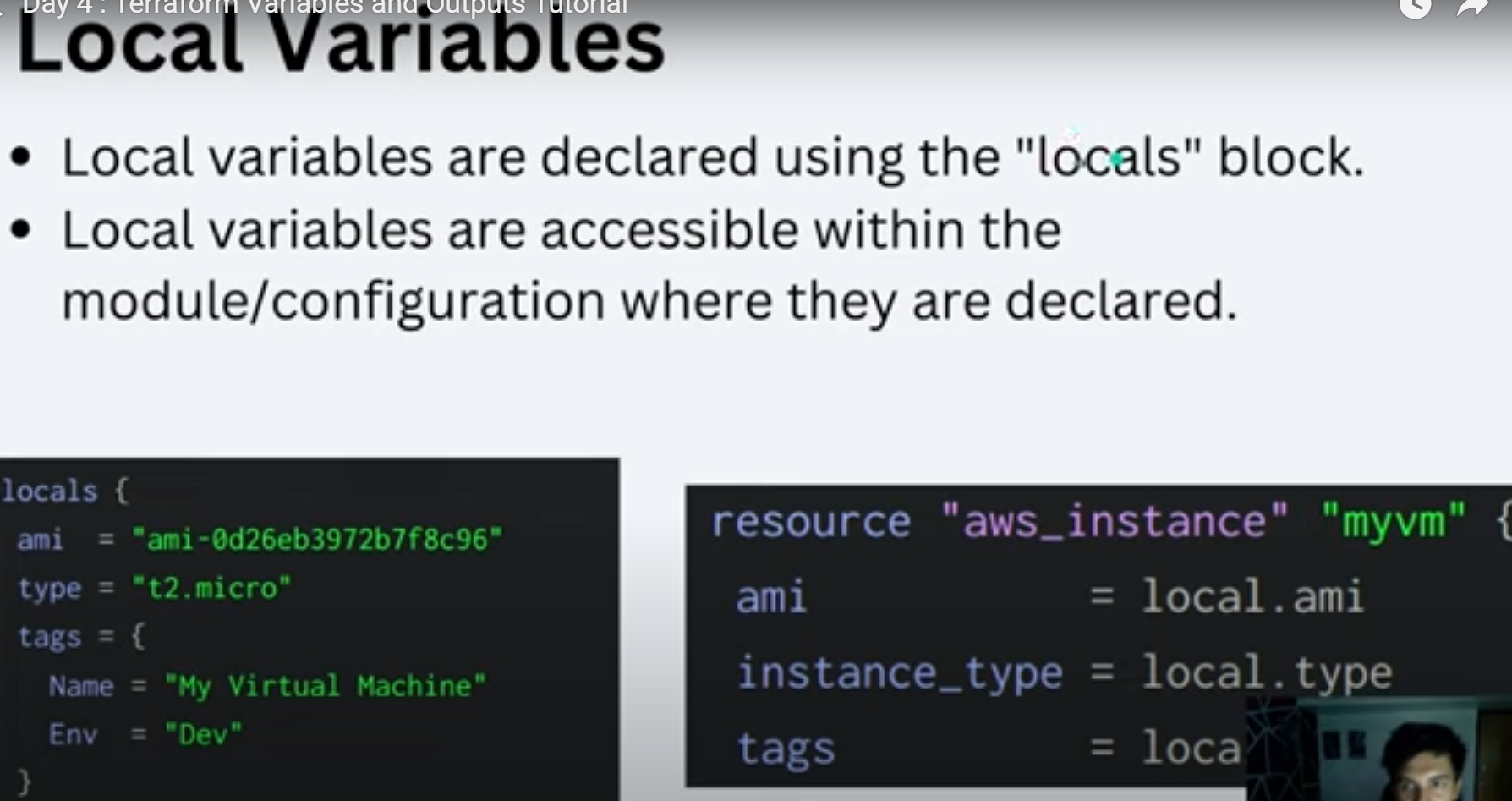
Created another resource for git in main.tf

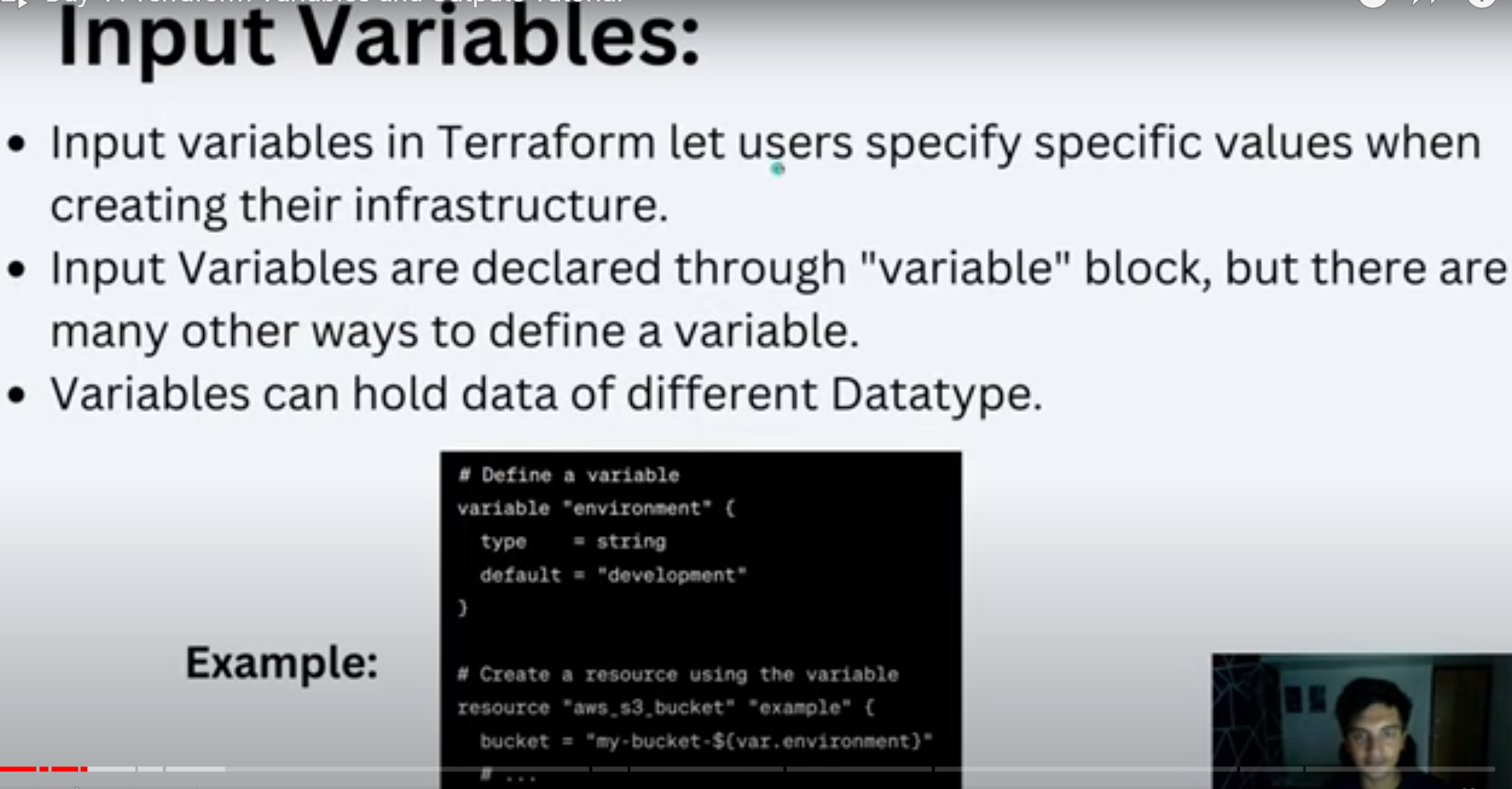


# Variables and Output

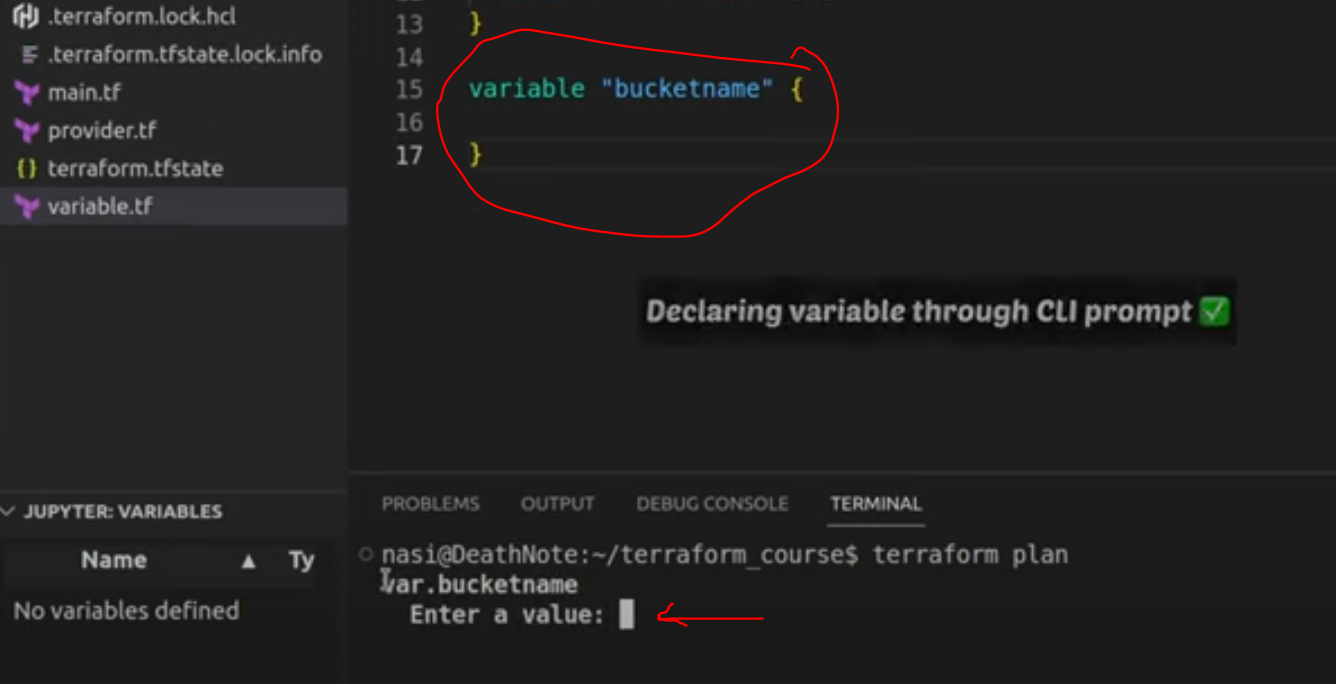
Types of variables

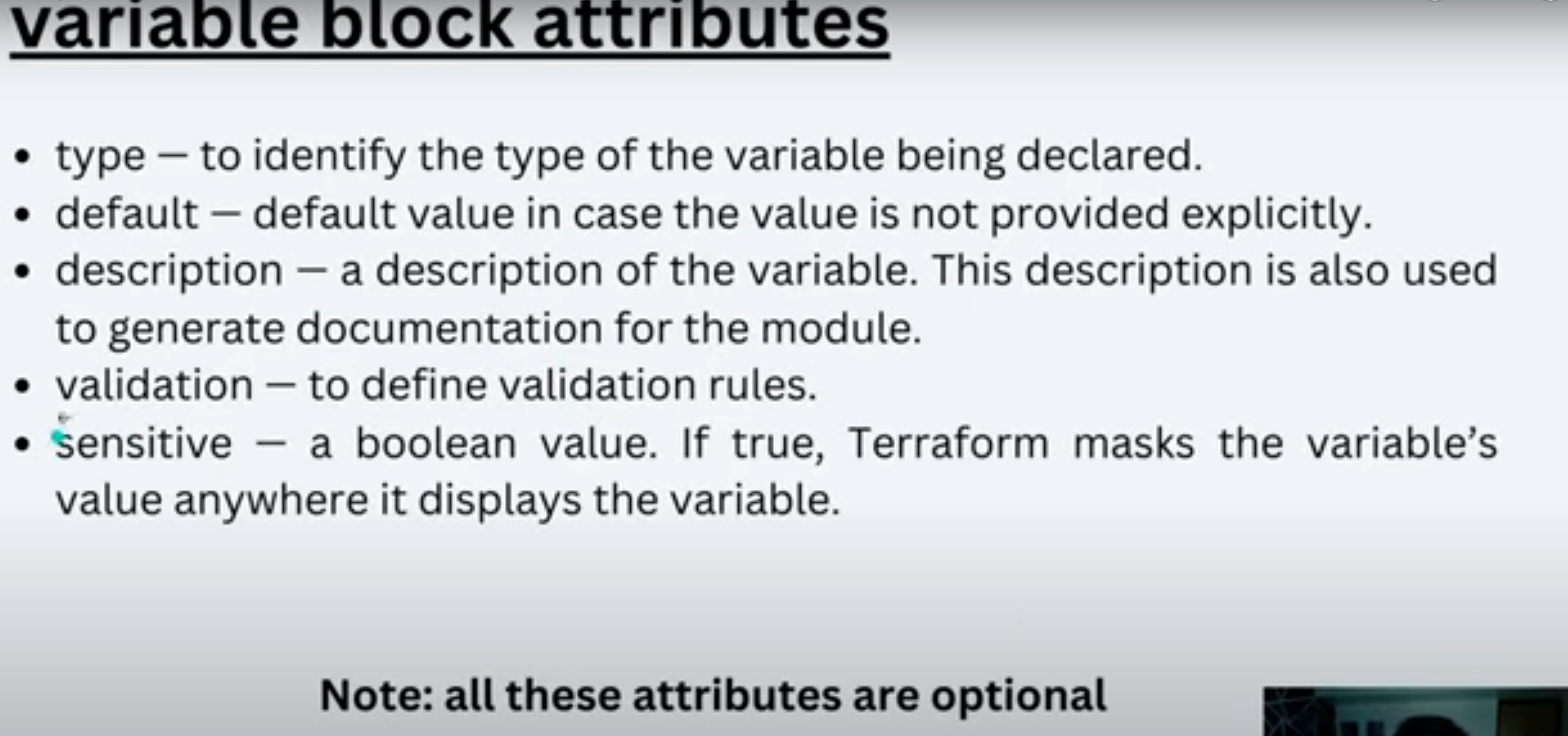




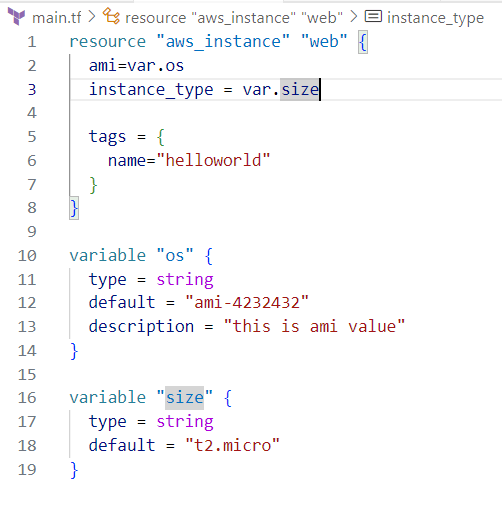


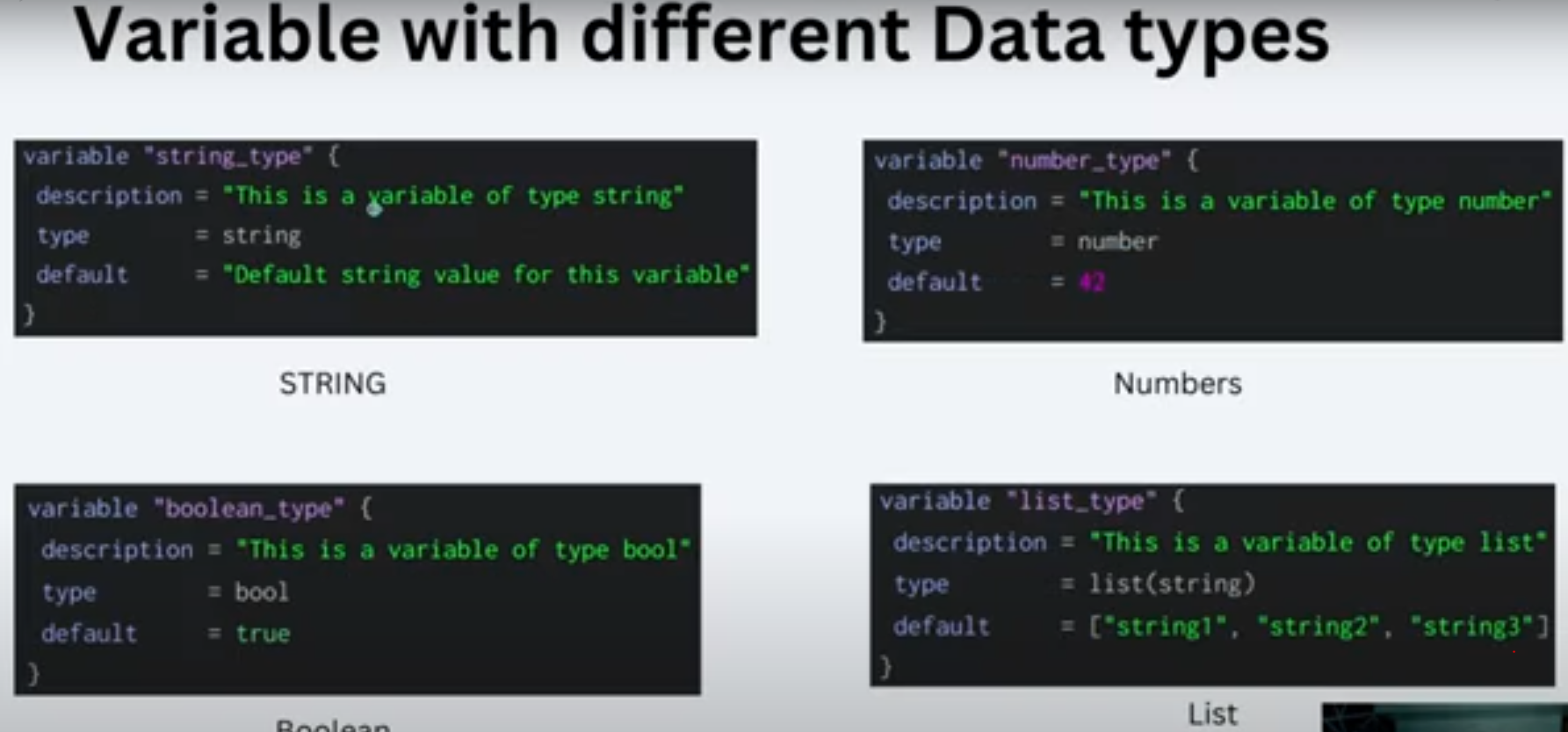
If we keep empty varible cli will ask its value

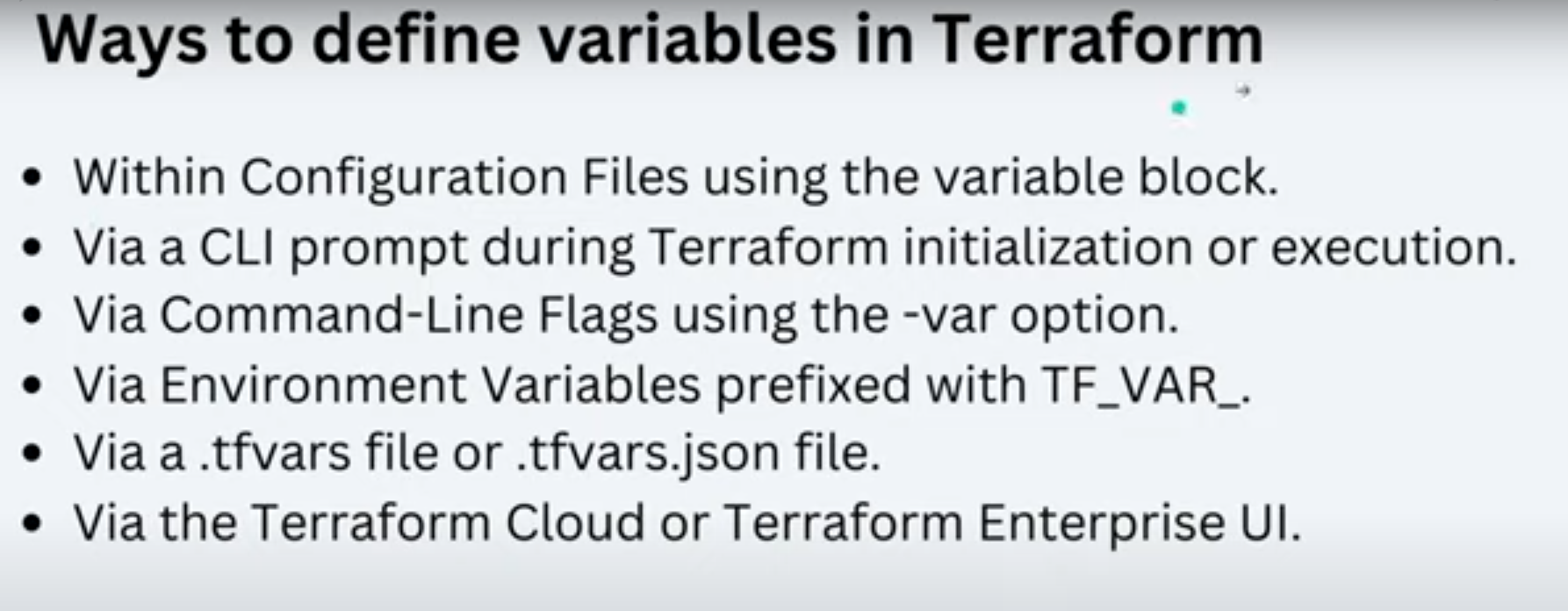




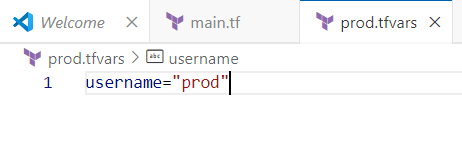
Sensitive true value will be visible in state file



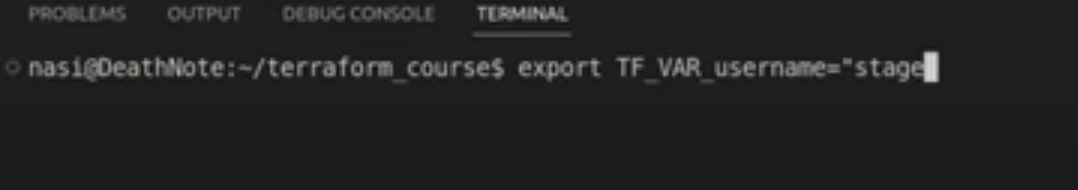


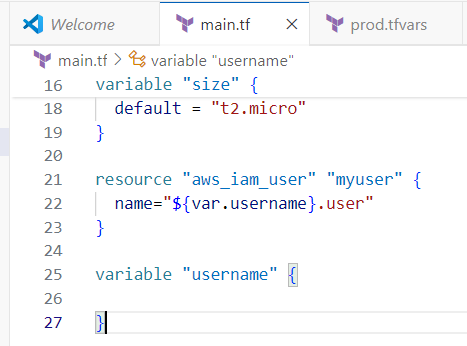


Using tfvar file to create variables

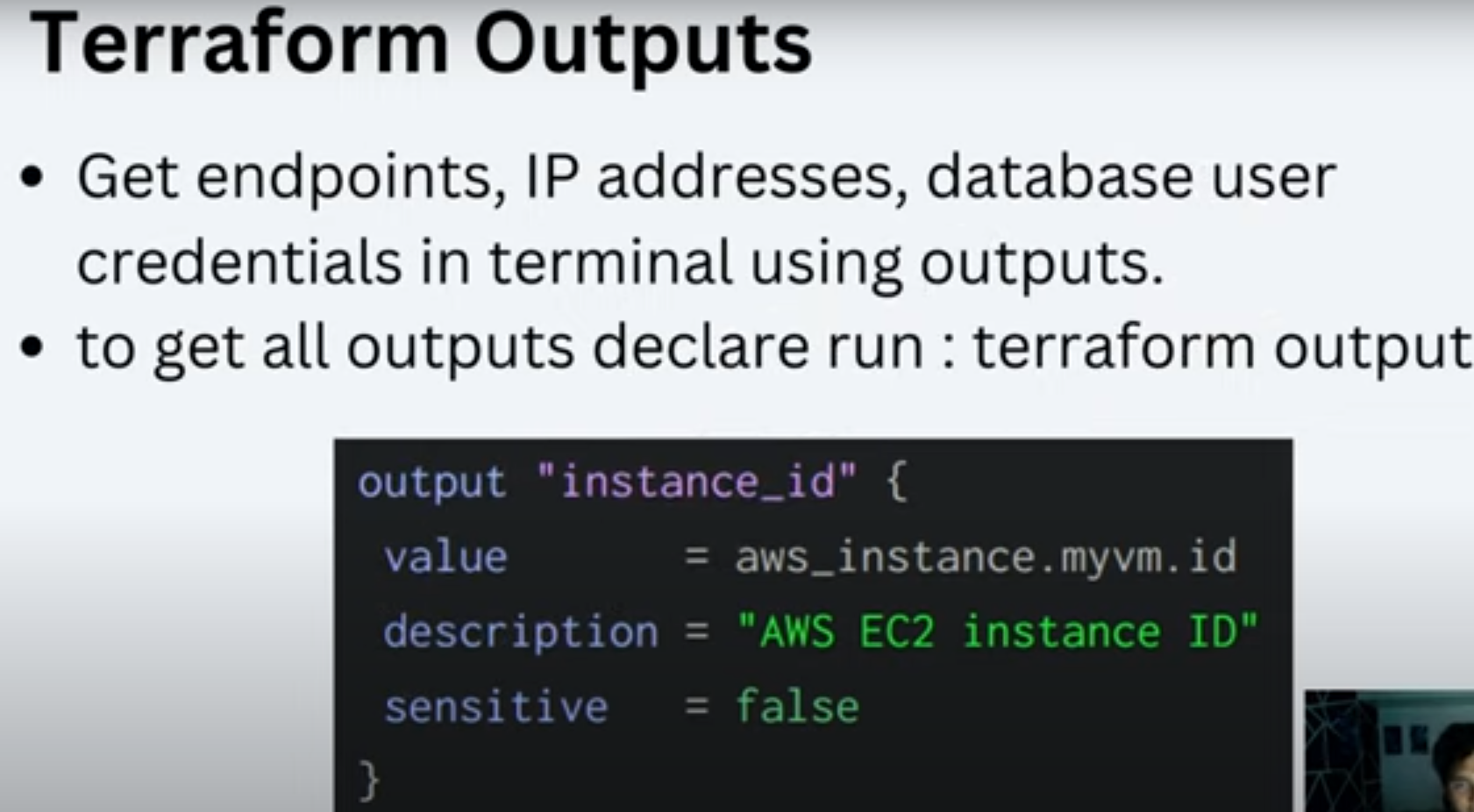


To create variable by enivirnment variable use below

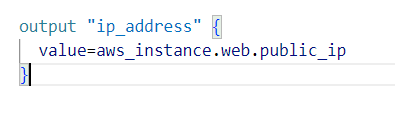




OUTPUTS:- suppose we need ip from EC2 resource we have created in tf in our cli so we use output for that



This will print output



Terraform output will list all outputs we have defined