Terraform

Workspace in Terraform

Terraform vault --> Scenario based on Terraform

Environments of the project:

Dev, QA, UAT, Pilot, Production

Environment refers to a platform or setup that’s required to run our application (Servers, Database, Storage, Networking,…)

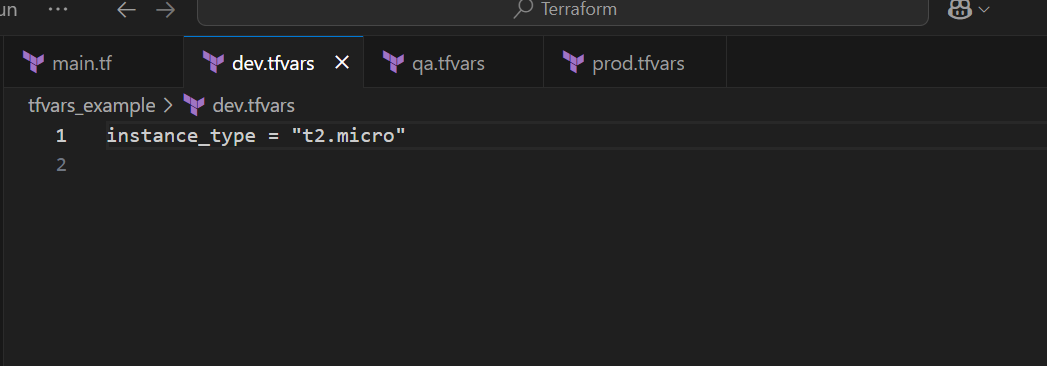
Generally we have multiple environments to run our projects

Say I want to use instance\_type = “t2.micro” for Dev, “t2.medium” for Production , like different instance\_types for different environments. If you run the Terraform script, a new instance is created. If “terraform.tfstate” file is already there, then when we re-apply or re-run the same scripts, will a new instance be created? No

I want to use the same script but want to create different resources for different environments.

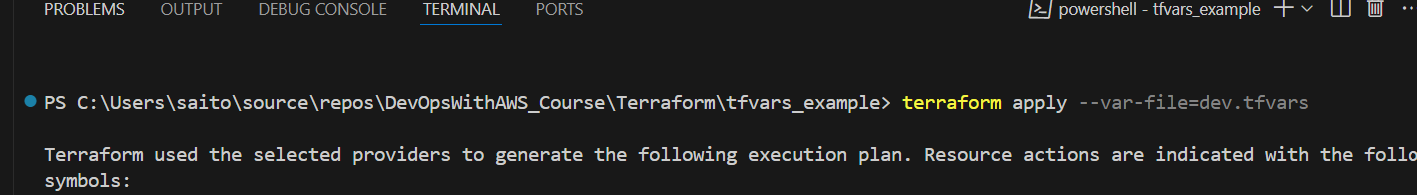
Then we have the concept .tfvars

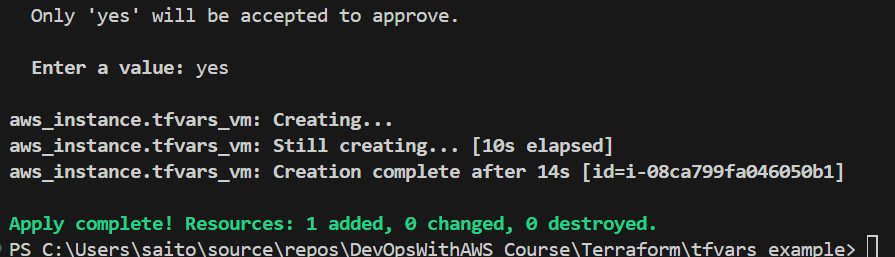
dev.tfvars, qa.tfvars, prod.tfvars --> different instance\_type for different environments

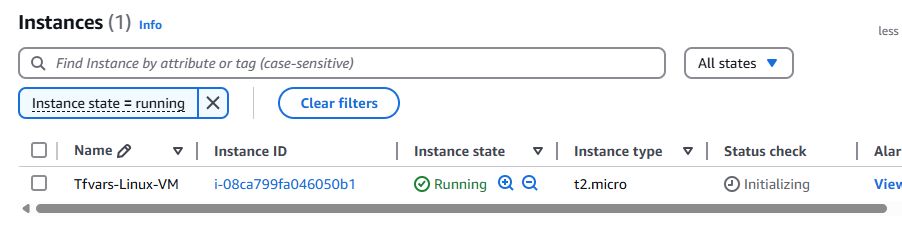


Dynamically pass the variable values

terraform apply --var-file=dev.tfvars





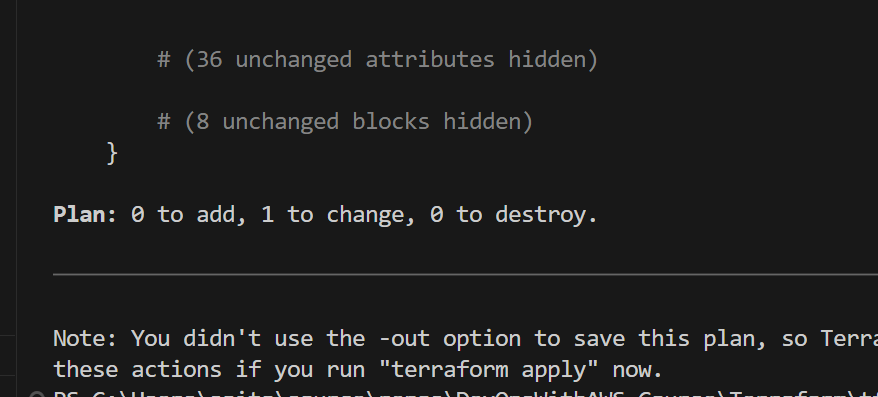


See instance\_type=”t2.micro”

We have created different ‘tfvars’ files for different environments but the state file is shared.

terraform plan --var-file=qa.tfvars

Then when I do a PLAN it is only trying to change the existing resource not ADD a new resource instead because of the common state file



What’s the solution? Workspace in Terraform

Dev environment --> t2.micro

QA environment --> t2.medium

Prod environment --> t2.xlarge

Now I want to create different state files for different environment, how’s this possible? **Workspace**

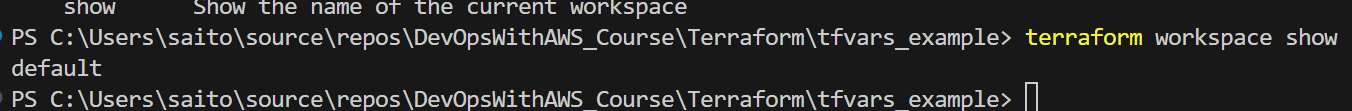
To manage infrastructure for Multiple environments, we will go with concept of Terraform workspace

If we go with Workspace concept then it wil maintain separate state files for every workspace

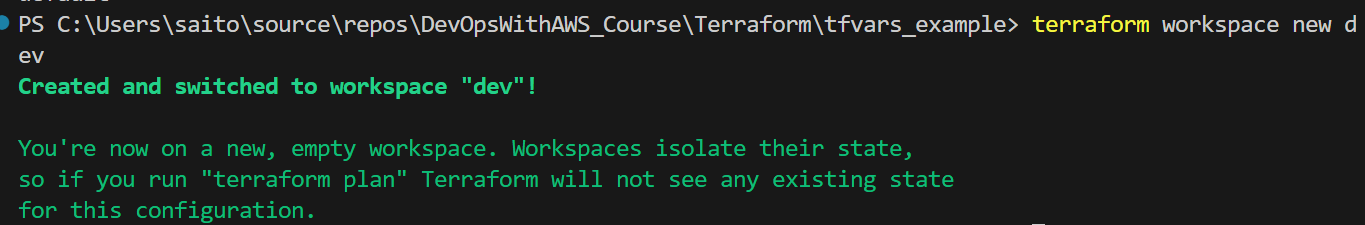
==> We can execute same script for multiple environments

Currently only one workspace is there: default

terraform workspace show

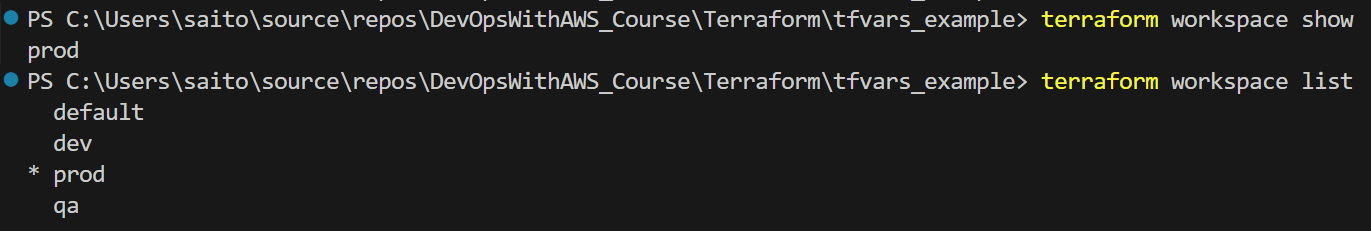


terraform workspace new dev --> create a new workspace for dev



terraform workspace new qa

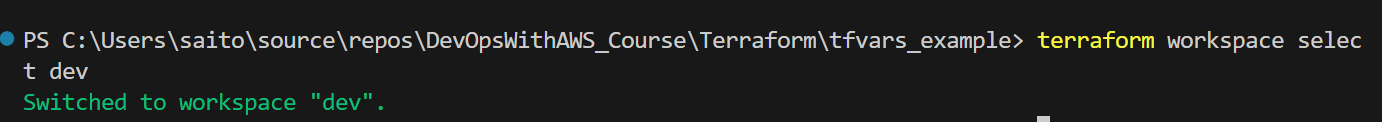
terraform workspace new prod

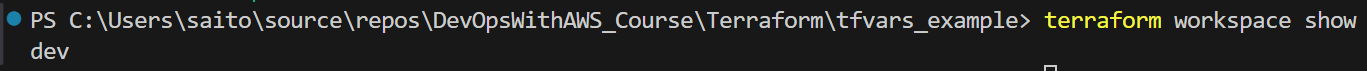


terraform workspace list

terraform workspace select dev

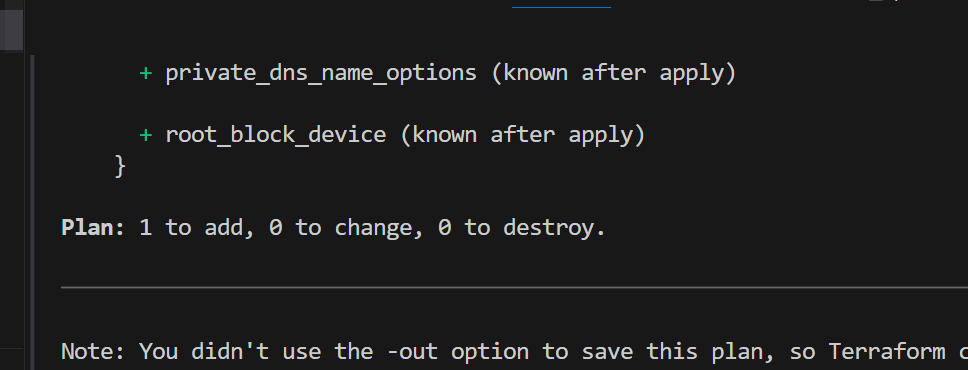
Switched to workspace "dev".



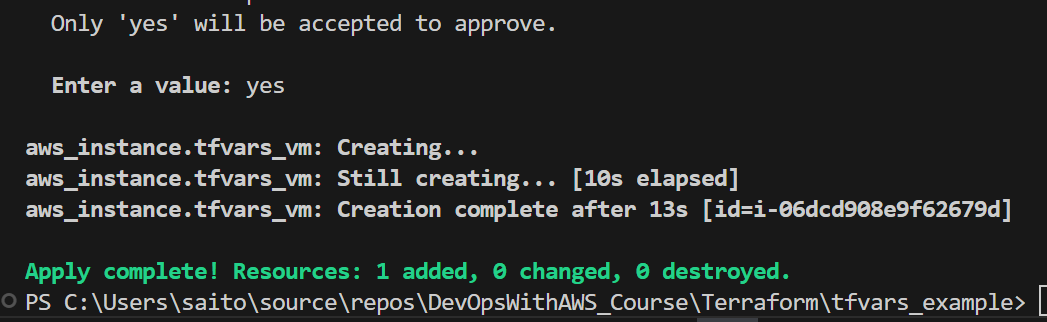


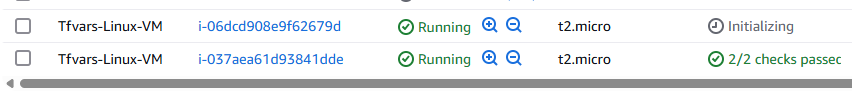
terraform plan --var-file=dev.tfvars

Now see Plan is 1 to add



terraform apply --var-file=dev.tfvars

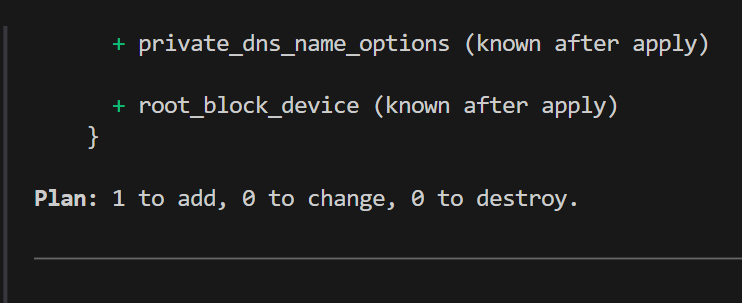




terraform workspace select qa

Switched to workspace "qa".

terraform plan --var-file=qa.tfvars



terraform workspace select dev

Switched to workspace "dev".

terraform destroy --auto-approve

Commands:

terraform workspace show --> shows current workspace

terraform workspace list --> show list of workspace

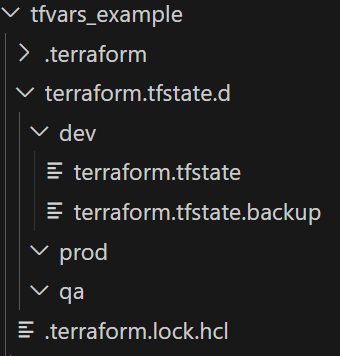
terraform workspace new dev --> create new ‘dev’ workspace

terraform workspace new qa --> creates new ‘qa’ workspace

terraform workspace select dev --> it will go to ‘dev’ workspace

terraform apply --var-file=dev.tfvars

We can see here it creates multiple State files



Infrastructure as a Code (IaC)

Terraform setup (Linux and Windows)

Terraform architecture

Terraform scripts (HCL)

Variables (Input variables, Output variables)

EC2 VM

S3 Buckets

IAM, VPC, RDS

Terraform modules:

State file, lockfile

Resource taint and untaint

Terraform workspace

Terraform vault