

general

Essential

- return terraform version
`terraform version`
- download and update root modules
`terraform get -update=true`
- open up a terraform interactive terminal
`terraform console`
- interact with terraform cloud
`terraform [login|logout]`
- check formatting
`terraform fmt`
- verify a configuration
`terraform validate`

initialize

Getting started

- initialize terraform in the current working dir
`terraform init`
- force plugin installation from a dir
`terraform init -plugin-dir=PATH`
- skip plugin installation
`terraform init -get-plugins=false`
- upgrade plugins and modules at init
`terraform init -upgrade`
- skip backend configuration
`terraform init -backend=false`
- update backend configuration
`terraform init -migrate-state -force-copy`
- skip plugin installation
`terraform init -get-plugins=false`

plan

Preview changes

- show the plan
`terraform plan`
- create a plan file for reference during apply
`terraform plan -out current.tfplan`
- show how a destroy command will effect the build
`terraform plan -destroy`
- target a specific resource for deployment
`terraform plan -target=RESOURCE`

apply

Make changes

- execute the plan created by the Terraform code
`terraform apply`
- execute a previously generated plan
`terraform apply current.tfplan`
- Execute with auto-approval (useful for automation)
`terraform apply -auto-approve`

outputs

Display values

- display any available outputs
`terraform output`
- show only a specific value
`terraform output VALUE`
- display output in JSON format
`terraform output -json`

destroy

Remove resources

- remove the plan created by the Terraform code
`terraform destroy`
- remove with auto-approval (useful for automation)
`terraform destroy -auto-approve`

workspaces

Work isolation

- create a workspace
`terraform workspace new myspace`
- select from workspaces
`terraform workspace select default`
- list all available workspaces
`terraform workspace list`

manage

Observe

- show all resources
`terraform state list`
- show details about a resource
`terraform state show RESOURCE`
- save current state to a local file
`terraform state pull > terraform.tfstate`
- show a dependency relationship (PNG)
`terraform graph | dot -Tpng > my_graph.png`