














## Terraform CLI — Quick View








### Initializing

	Command	Description	Example
	terraform init	Initialize working directory, download providers/modules, configure backend. Required before any plan/apply.	terraform init
	terraform init -backend-config	Load backend config file or key=value pairs; supports S3, GCS, TFC.	terraform init -backend-config=prod.s3.tfbackend
	terraform init -upgrade	Re-check registry for newer allowable provider/module versions and upgrade if needed.	terraform init -upgrade
	terraform init -migrate-state	Migrate existing state to new backend (S3→TFC, etc.).	terraform init -migrate-state
	terraform init -reconfigure	Ignore cached backend settings; force new re-initialization.	terraform init -reconfigure
	terraform get	Download modules referenced by root module.	terraform get -update
	terraform metadata	Show internal metadata (Terraform 1.5+).	terraform metadata













### Planning

	Command	Description	Example
	terraform plan	Preview infra changes; shows additions, updates, destroys. Essential before apply.	terraform plan
	terraform plan -out=<file_name>	Save a binary plan file for reproducible apply.	terraform plan -out=prod.tfplan
	terraform plan -refresh-only	Detect drift without modifying state.	terraform plan -refresh-only
	terraform plan -var '<key>=<value>' / -var-file	Pass input variables from CLI or file.	terraform plan -var-file=prod.tfvars
	terraform plan -replace <resource>	Show the plan to Force recreation of specific resource instances.	terraform plan -replace=aws_instance.web
	terraform plan -target <resource>	Create Plan for specific module or resource. Partial planning—can miss dependencies; only for recovery/emergencies.	terraform plan -target=aws_s3_bucket.logs





### Apply & Destroy

	Command	Description	Example
	terraform apply	Apply the plan and make real cloud changes.	terraform apply
	terraform apply <planfile>	Execute saved plan exactly as generated; prevents drift.	terraform apply prod.tfplan
	terraform apply -auto-approve	Skip approval for automation pipelines.	terraform apply -auto-approve
	terraform apply -target <resource>	To create or update only specific resource	terraform apply -target=aws_s3_bucket.logs
	terraform apply -replace <resource>	Force the replacement of a specific resource ( destroyed/recreated )	terraform apply -replace=aws_instance.web -replace=module.app.aws_autoscaling_group.main
	terraform destroy	Destroy entire infrastructure; requires caution.	terraform destroy
	terraform destroy -target <resource>	Destroy the specific resource. ( considered a niche/edge-use feature)	terraform destroy -target=module.storage.aws_s3_bucket.logs






### State Management

	Command	Description	Example
	terraform show	Show the current state in human-readable form	terraform show
	terraform show <file>	Show a human-readable state or plan from mentioned file	terraform show plan.tfplan
	terraform show -json <file>	Full JSON representation of plan/state.	terraform show -json plan.tfplan
	terraform state list	List all resources tracked in state.	terraform state list
	terraform state show <addr>	Show attributes of resource in state.	terraform state show aws_vpc.main
	terraformstate mv <source> <destination>	Move resource addresses within state, useful after refactoring.	terraform state mv aws_iam_user.dev module.users.aws_iam_user.dev
	terraform state rm <resource>	Remove resource from state without deleting it—risky.	terraform state rm aws_s3_bucket.orphan
	terraform state replace-provider <from> <to>	Change provider references in state—required after upgrading providers.	terraform state replace-provider registry.terraform.io/-/aws registry.terraform.io/hashicorp/aws
	terraform state pull	Download remote state to stdout.	terraform state pull
	terraform force-unlock <lock_id>	Release a stuck remote backend lock.	terraform force-unlock -force 1234
	terraform import <resource> <remote_id>	Map existing cloud resources into Terraform state.	terraform import aws_instance.web i-012345678
	terraform refresh (deprecated)	Updates state from cloud, but replaced by plan/apply -refresh-only.	terraform refresh

### Outputs

	Command	Description	Example
	terraform output	Show all output values from root module.	terraform output
	terraform output <name>	Show a specific output Value	terraform output db_password
	terraform output -json	Structured machine output.	terraform output -json
	terraform output -raw <name>	Print raw value (string/number/bool).	terraform output -raw db_password
	terraform show -json	Full JSON representation of plan/state.	terraform show -json plan.tfplan








### Workspaces

	Command	Description	Example
	terraform workspace list	List workspaces. The Workspace having a * sign it the current workspace	terraform workspace list
	terraform workspace show	Show the name of current workspace	terraform workspace show
	terraform workspace new <name>	Create isolated state.	terraform workspace new dev
	terraform workspace select <name>	Switch workspace.	terraform workspace select prod
	terraform workspace delete <name>	Delete workspace.	terraform workspace delete dev

### Providers

	Command	Description	Example
	terraform providers	Show all providers required.	terraform providers
	terraform providers mirror	Create local mirror repo for offline use.	terraform providers mirror ./mirror
	terraform providers schema -json	Output full provider resource/data source schema.	terraform providers schema -json > schema.json
	terraform login/logout	Authenticate to Terraform Cloud or private registries.	terraform login app.terraform.io

### Utilities & Tools

	Command	Description	Example
	terraform console	Evaluate expressions interactively.	terraform console
	terraform graph	Output Graphviz dependency graph.	terraform graph   dot -Tpng > graph.png
	terraform fmt	Reformat code to canonical HCL style.	terraform fmt -recursive
	terraform validate	Validate syntax and provider config.	terraform validate
	terraform modules	Show declared modules in working directory.	terraform modules
	terraform -chdir (global)	Run a Terraform command from another directory.	terraform -chdir=prod plan
	terraform -help	Show help for any subcommand.	terraform plan -help

For More on this see my Git repository:

<https://github.com/mohakhan03uk/terraform/>