

```

harsh@Harsh:~$ sudo apt-get install terraform
[sudo] password for harsh:
Reading package lists... Done
harsh@Harsh:~$ terraform -help
Command 'terraform' not found, but can be installed with:
sudo snap install terraformhelp
harsh@Harsh:~$ terraform --help
harsh@Harsh:~$ wget -O- https://apt.releases.hashicorp.com/gpg | \
rmor | \
sudo tee /usr/share/keyrings/hashicorp-archive-keyring.gpg > /dev/null
--2024-03-07 18:45:56-- https://apt.releases.hashicorp.com/gpg
Resolving apt.releases.hashicorp.com (apt.releases.hashicorp.com)... 18.245.96.102, 18.245.96.14, 18.245.96.72, ...
Connecting to apt.releases.hashicorp.com (apt.releases.hashicorp.com)[18.245.96.102]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3980 (3.9K) [binary/octet-stream]
Saving to: 'STDOUT'

-
100%[=====] 3.89K --.-KB/s in 0s

2024-03-07 18:45:56 (730 MB/s) - written to stdout [3980/3980]

harsh@Harsh:~$ gpg --no-default-keyring \
> --keyring /usr/share/keyrings/hashicorp-archive-keyring.gpg \
> --fingerprint
gpg: directory '/home/harsh/.gnupg' created
gpg: /home/harsh/.gnupg/trustdb.gpg: trustdb created
/usr/share/keyrings/hashicorp-archive-keyring.gpg
-----
pub  rsa4096 2023-01-10 [SC] [expires: 2028-01-09]
     798A ECC6 4E5C 1542 8C0E 42EE A416 FC0C A021 E701
uid          [ unknown] HashiCorp Security (HashiCorp Package Signing) <security+packaging@hashicorp.com>
sub  rsa4096 2023-01-10 [S] [expires: 2028-01-09]

harsh@Harsh:~$ echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] \
//apt.> https://apt.releases.hashicorp.com $(lsb_release -cs) main" | \
> sudo tee /etc/apt/sources.list.d/hashicorp.list
deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com jammy main
harsh@Harsh:~$ sudo apt update
Hit:1 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:4 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Get:5 https://apt.releases.hashicorp.com jammy InRelease [12.9 kB]
Hit:6 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:7 https://apt.releases.hashicorp.com jammy/main amd64 Packages [121 kB]
Get:8 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [27.6 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1239 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1458 kB]
Get:11 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [222 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1523 kB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [253 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [848 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [162 kB]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.1 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [283 kB]
Get:18 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1557 kB]
Get:19 http://archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [259 kB]
Get:20 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1054 kB]
Get:21 http://archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [238 kB]
Get:22 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [42.1 kB]
Fetched 9614 kB in 3s (3399 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
30 packages can be upgraded. Run 'apt list --upgradable' to see them.
W: https://download.docker.com/linux/ubuntu/dists/jammy/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
harsh@Harsh:~$ sudo apt-get install terraform
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  terraform
0 upgraded, 1 newly installed, 0 to remove and 30 not upgraded.
Need to get 26.9 MB of archives.
After this operation, 84.5 MB of additional disk space will be used.
Get:1 https://apt.releases.hashicorp.com jammy/main amd64 terraform amd64 1.7.4-1 [26.9 MB]
Fetched 26.9 MB in 1s (21.1 MB/s)

```

```
harsh@Harsh: ~
The following NEW packages will be installed:
  terraform
0 upgraded, 1 newly installed, 0 to remove and 30 not upgraded.
Need to get 26.9 MB of archives.
After this operation, 84.5 MB of additional disk space will be used.
Get:1 https://apt.releases.hashicorp.com jammy/main amd64 terraform amd64 1.7.4-1 [26.9 MB]
Fetched 26.9 MB in 1s (21.1 MB/s)
Selecting previously unselected package terraform.
(Reading database ... 83603 files and directories currently installed.)
Preparing to unpack .../terraform_1.7.4-1_amd64.deb ...
Unpacking terraform (1.7.4-1) ...
Setting up terraform (1.7.4-1) ...
harsh@Harsh:~$ terraform -help
Usage: terraform [global options] <subcommand> [args]

The available commands for execution are listed below.
The primary workflow commands are given first, followed by
less common or more advanced commands.

Main commands:
  init      Prepare your working directory for other commands
  validate  Check whether the configuration is valid
  plan      Show changes required by the current configuration
  apply     Create or update infrastructure
  destroy   Destroy previously-created infrastructure

All other commands:
  console   Try Terraform expressions at an interactive command prompt
  fmt       Reformat your configuration in the standard style
  force-unlock Release a stuck lock on the current workspace
  get       Install or upgrade remote Terraform modules
  graph     Generate a Graphviz graph of the steps in an operation
  import    Associate existing infrastructure with a Terraform resource
  login     Obtain and save credentials for a remote host
  logout    Remove locally-stored credentials for a remote host
  metadata  Metadata related commands
  output    Show output values from your root module
  providers Show the providers required for this configuration
  refresh   Update the state to match remote systems
  show      Show the current state on a saved plan
  state     Advanced state management
  taint     Mark a resource instance as not fully functional
  test      Execute integration tests for Terraform modules
```

The screenshot shows a VS Code editor window with a file named `main.tf` open. The file contains a Terraform configuration for a Docker container. The configuration includes a `provider "docker"` block, a `resource "docker_image" "nginx"` block, and a `resource "docker_container" "nginx"` block. The `docker_image` resource is configured with `name = "nginx"` and `keep_locally = false`. The `docker_container` resource is configured with `image = docker_image.nginx.image_id`, `name = "tutorial"`, and `ports` with `internal = 80` and `external = 8001`.

The terminal window at the bottom shows the output of the `terraform init` command. The output indicates that the backend is being initialized, provider plugins are being installed, and the configuration is being validated. The output also shows the installation of the `docker` provider and the creation of the `terraform.tfstate` file.

```
main.tf
1 terraform {
2   backend "local" {}
3 }
4
5 provider "docker" {}
6
7 resource "docker_image" "nginx" {
8   name = "nginx"
9   keep_locally = false
10 }
11
12 resource "docker_container" "nginx" {
13   image = docker_image.nginx.image_id
14   name = "tutorial"
15   ports {
16     internal = 80
17     external = 8001
18   }
19 }
20
21
22
23
24
```

```
harsh@harsh:~/learn-terraform-docker-container$ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding kreuzwerker/docker versions matching "~> 3.0.1"...
- Installing kreuzwerker/docker v3.0.2...
- Installed kreuzwerker/docker v3.0.2 (self-signed, key ID BD080C4571C6104C)

Partner and community providers are signed by their developers.
If you'd like to know more about provider signing, you can read about it here:
https://www.terraform.io/docs/cli/plugins/signing.html
```



