

**Aim:- To Build, change, and destroy AWS / GCP /Microsoft Azure/
DigitalOcean infrastructure Using Terraform. (S3 bucket or Docker) fdp**

```
Microsoft Windows [Version 10.0.22631.3958]  
(c) Microsoft Corporation. All rights reserved.
```

```
C:\Users\Santosh Sawant>docker --version  
Docker version 27.0.3, build 7d4bcd8
```

```
C:\Users\Santosh Sawant>
```

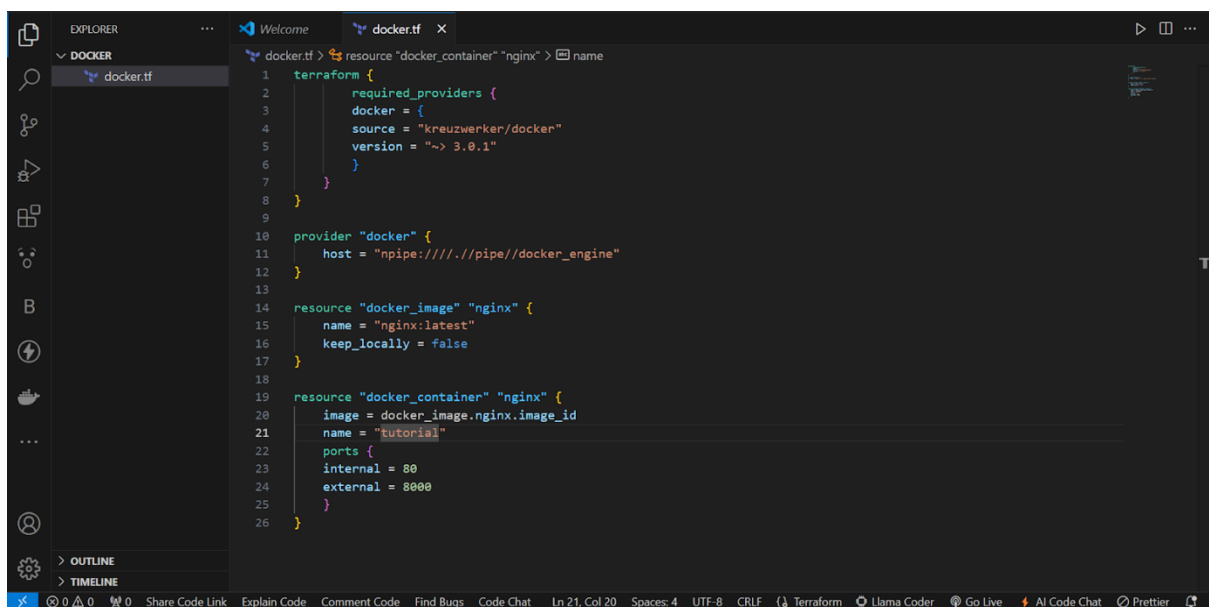
Code:-

```
terraform {  
    required_providers {  
        docker = {  
            source = "kreuzwerker/docker"  
            version = "~> 3.0.1"  
        }  
    }  
}  
  
provider "docker" {  
    host = "npipe:////./pipe//docker_engine"  
}  
  
resource "docker_image" "nginx" {  
    name = "nginx:latest"}
```

```

    keep_locally = false
  }
  resource "docker_container" "nginx" {
    image = docker_image.nginx.image_id
    name = "tutorial"
    ports {
      internal = 80
      external = 8000
    }
  }
}

```



Output:-

Terraform init:-

```

● PS C:\Terraform scripts\Docker> 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
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

```

Terraform plan:-

```

● PS C:\Terraform scripts\Docker> terraform plan

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following
symbols:
+ create

Terraform will perform the following actions:

# docker_container.nginx will be created
+ resource "docker_container" "nginx" {
  + attach                = false
  + bridge                = (known after apply)
  + command               = (known after apply)
  + container_logs        = (known after apply)
  + container_read_refresh_timeout_milliseconds = 15000
}

```

Terraform apply:-

```

● PS C:\Terraform scripts\Docker> terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following
symbols:
+ create

Terraform will perform the following actions:

# docker_container.nginx will be created
+ resource "docker_container" "nginx" {
  + attach                = false
  + bridge                = (known after apply)
  + command               = (known after apply)
  + container_logs        = (known after apply)
  + container_read_refresh_timeout_milliseconds = 15000
}

```

Terraform destroy:-

```

PS C:\Terraform scripts\Docker> terraform destroy
docker_image.nginx: Refreshing state... [id=sha256:5ef79149e0ec84a7a9f9284c3f91aa3c20608f8391f5445eabe92ef07dbda03cnginx:latest]
docker_container.nginx: Refreshing state... [id=96d0c83b0b6bcf2351b8f717d11d4ee0a27d765760211da08e1ced043154578c]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following
symbols:
  - destroy

Terraform will perform the following actions:

# docker_container.nginx will be destroyed
- resource "docker_container" "nginx" {
  - attach          = false -> null
  - command         = [
    - "nginx",
  ]
}

```

Before using terraform commands:-

```

PS C:\Terraform scripts\Docker> docker images

```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
ubuntu	latest	edbfe74c41f8	3 weeks ago	78.1MB

```

PS C:\Terraform scripts\Docker> docker container list

```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

After using terraform commands:-

```

PS C:\Terraform scripts\Docker> docker images

```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	5ef79149e0ec	10 days ago	188MB
ubuntu	latest	edbfe74c41f8	3 weeks ago	78.1MB

```

PS C:\Terraform scripts\Docker> docker container list

```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
96d0c83b0b6b	5ef79149e0ec	"/docker-entrypoint..."	6 minutes ago	Up 6 minutes	0.0.0.0:8000->80/tcp	tutorial