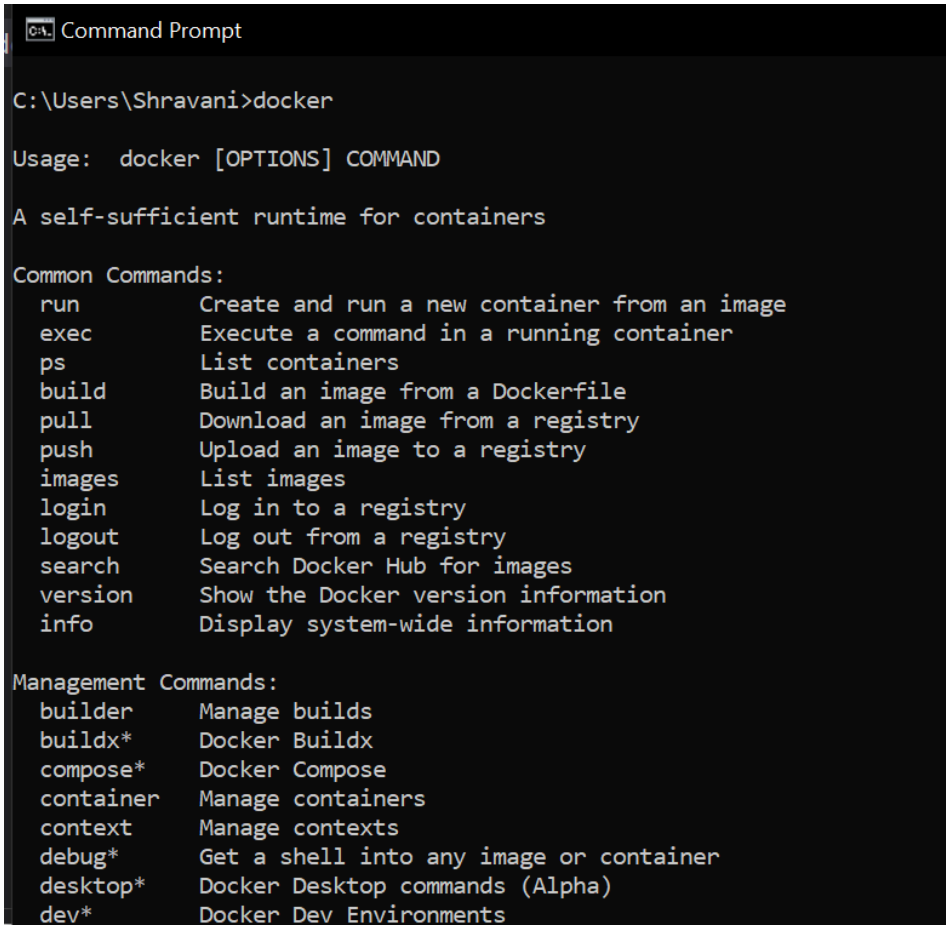
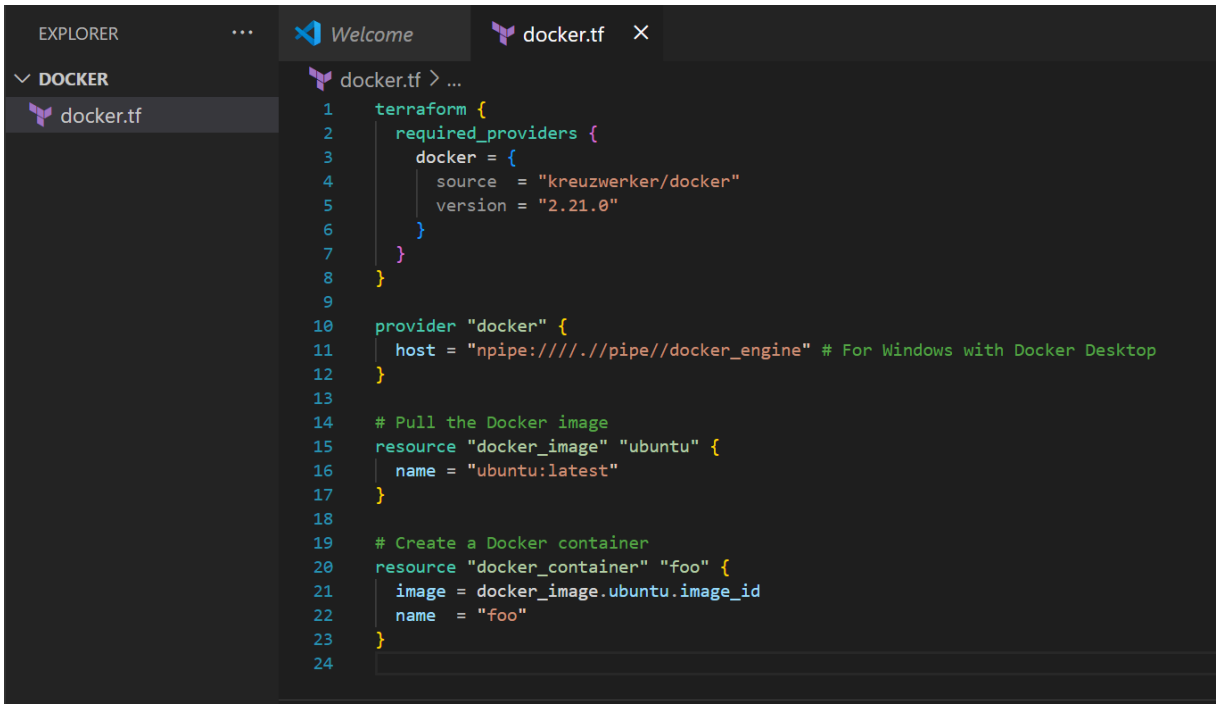
ADVANCED DEVOPS EXP 6

AIM: Creating docker image using terraform

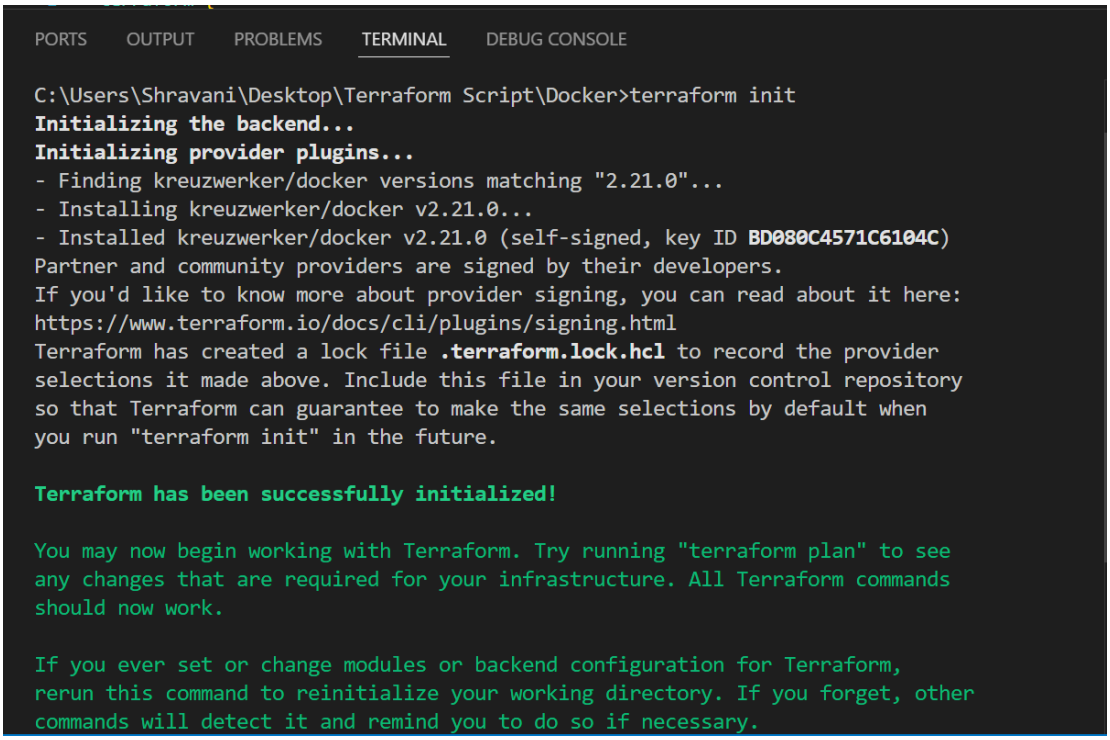
Step 1: Check the docker functionality



Step 2: Firstly create a new folder named ‘Docker’ in the ‘TerraformScripts’ folder. Then create a new docker.tf file using Atom editor and write the followingcontents into it to create a Ubuntu Linux container.



Step 3: Execute Terraform Init command to initialize the resources



Step 4: Execute Terraform plan to see the available resources

```
PORTS   OUTPUT   PROBLEMS   TERMINAL   DEBUG CONSOLE

C:\Users\Shravani\Desktop\Terraform Script\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.foo will be created
  + resource "docker_container" "foo" {
      + attach           = false
      + bridge           = (known after apply)
      + command          = (known after apply)
      + container_logs   = (known after apply)
      + entrypoint       = (known after apply)
      + env              = (known after apply)
      + exit_code        = (known after apply)
      + gateway          = (known after apply)
      + hostname         = (known after apply)
      + id               = (known after apply)
      + image            = (known after apply)
      + init             = (known after apply)
      + ip_address       = (known after apply)
```

```
PORTS   OUTPUT   PROBLEMS   TERMINAL   DEBUG CONSOLE

      + healthcheck (known after apply)

      + labels (known after apply)
    }

  # docker_image.ubuntu will be created
  + resource "docker_image" "ubuntu" {
      + id          = (known after apply)
      + image_id    = (known after apply)
      + latest      = (known after apply)
      + name        = "ubuntu:latest"
      + output      = (known after apply)
      + repo_digest = (known after apply)
    }

Plan: 2 to add, 0 to change, 0 to destroy.

---

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to
take exactly these actions if you run "terraform apply" now.

C:\Users\Shravani\Desktop\Terraform Script\Docker>
```

Step 5: Execute Terraform apply to apply the configuration, which will automatically create and run the Ubuntu Linux container based on our configuration. Using command : “terraform apply”

Docker images, Before Executing Apply step:

```
C:\Users\Shravani\Desktop\Terraform Script\Docker>docker images
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE
```

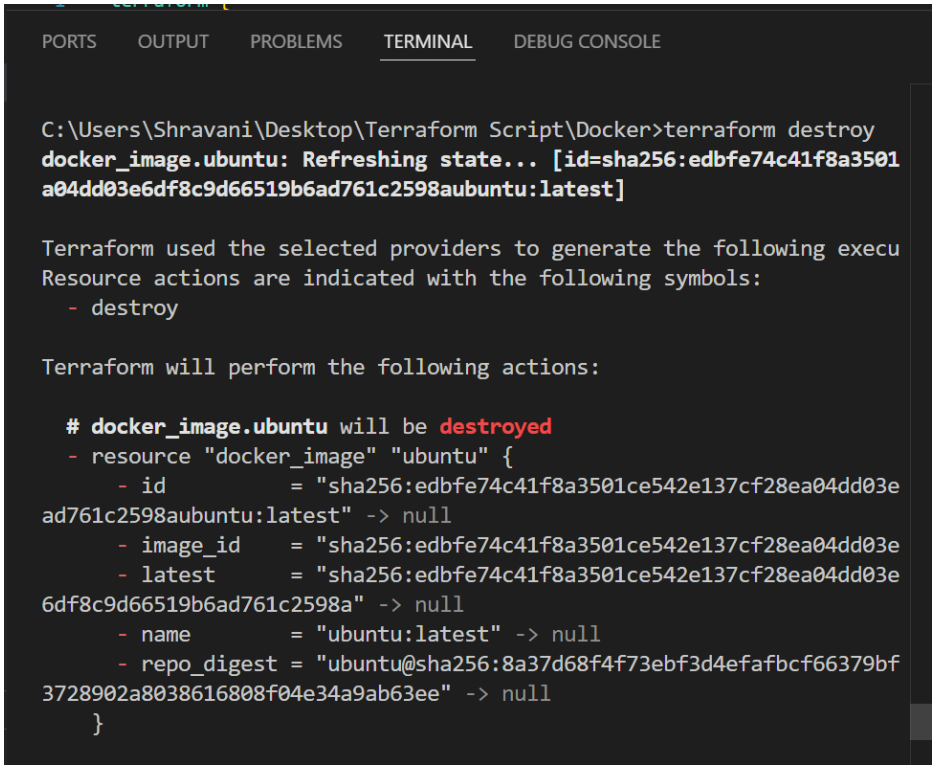
terraform apply

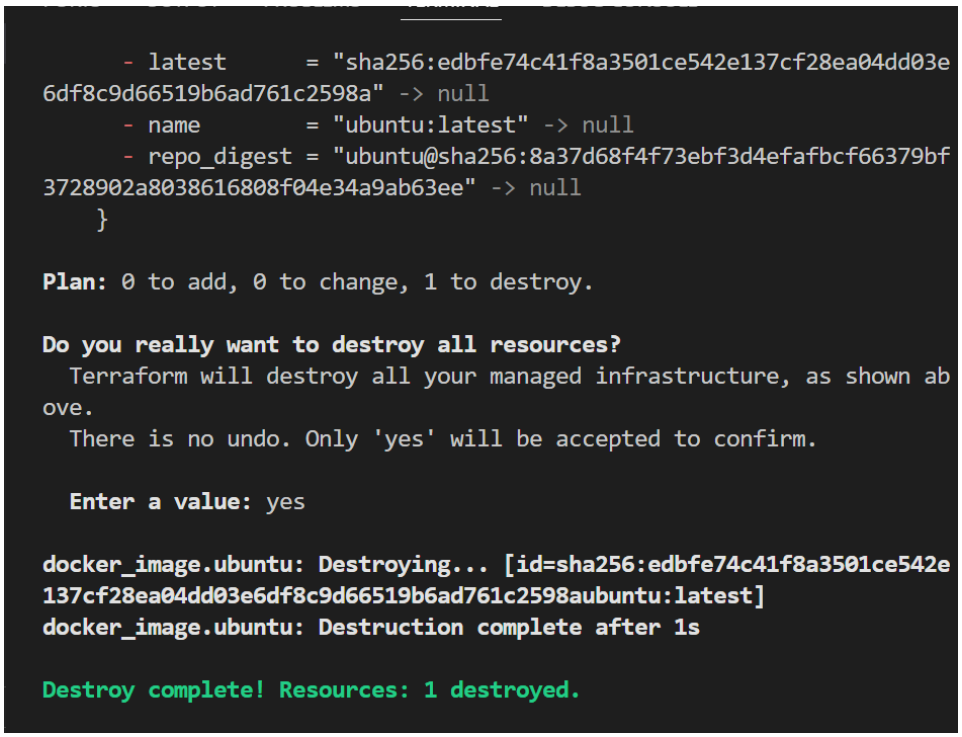
```
PORTS   OUTPUT   PROBLEMS   TERMINAL   DEBUG CONSOLE

C:\Users\Shravani\Desktop\Terraform Script\Docker>terraform apply
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28e
3e6df8c9d66519b6ad761c2598aubuntu:latest]

Note: Objects have changed outside of Terraform

Terraform detected the following changes made outside of Terraform since the last
"terraform apply" which may have affected this plan:

  # docker_image.ubuntu has been deleted
  - resource "docker_image" "ubuntu" {
        id          = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6
2598aubuntu:latest"
      - image_id    = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6
2598a" -> null
        name        = "ubuntu:latest"
        # (2 unchanged attributes hidden)
    }


Unless you have made equivalent changes to your configuration, or ignored the relev
attributes using ignore_changes, the following plan may include actions to undo or
respond to these changes.
```

Docker images, After Executing Apply step:

```
C:\Users\Shravani\Desktop\Terraform Script\Docker>docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
ubuntu       latest    edbfe74c41f8   3 weeks ago   78.1MB
```

Step 6: Execute Terraform destroy to delete the configuration, which will automatically delete the Ubuntu Container.





Docker images After Executing Destroy step

```
Destroy complete! Resources: 1 destroyed.

C:\Users\Shravani\Desktop\Terraform Script\Docker>docker images
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE
```