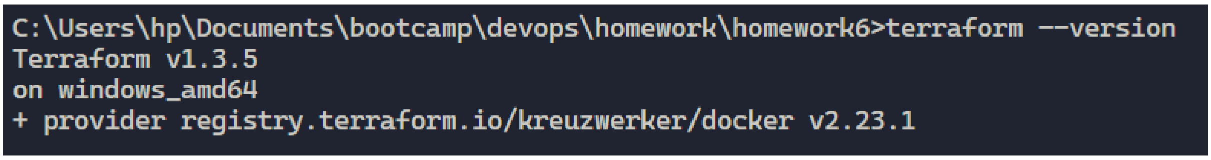
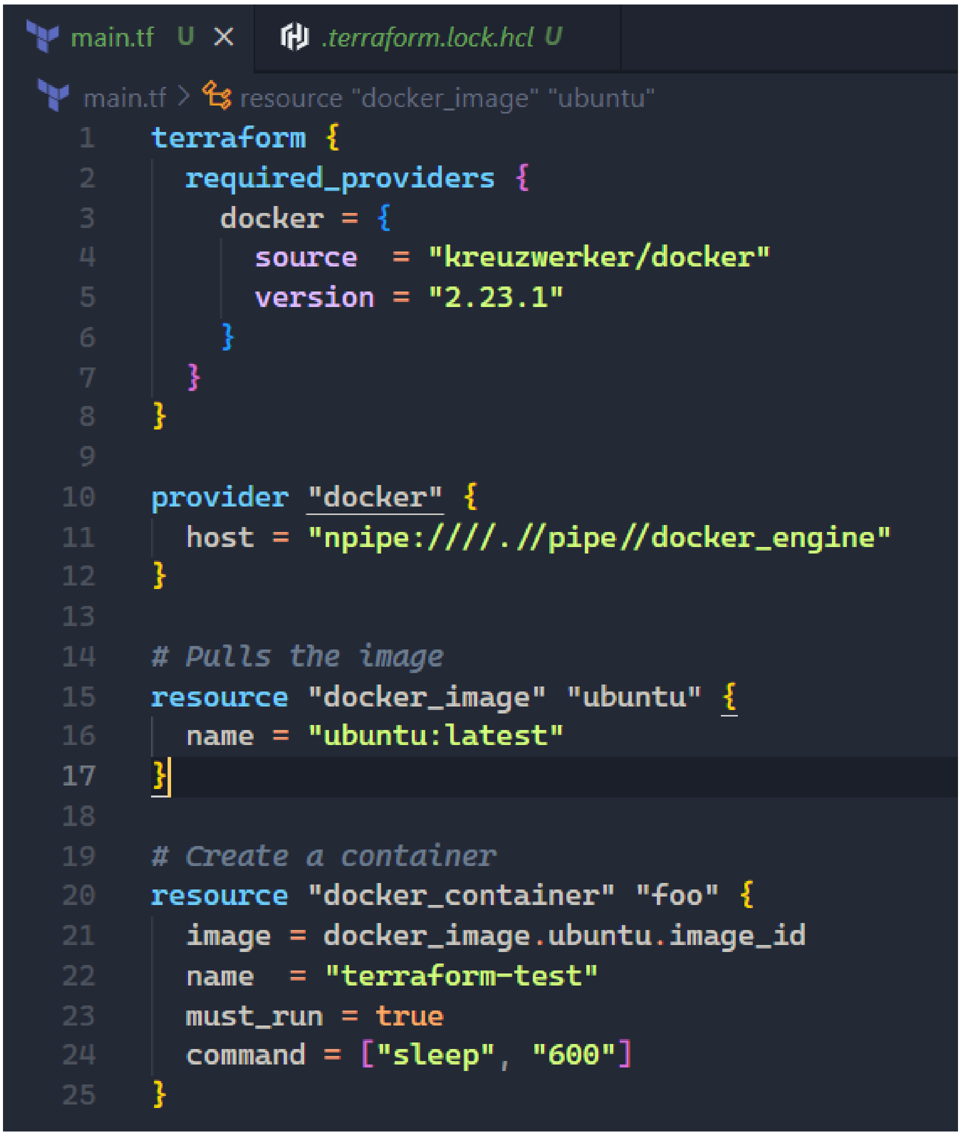
Terraform install



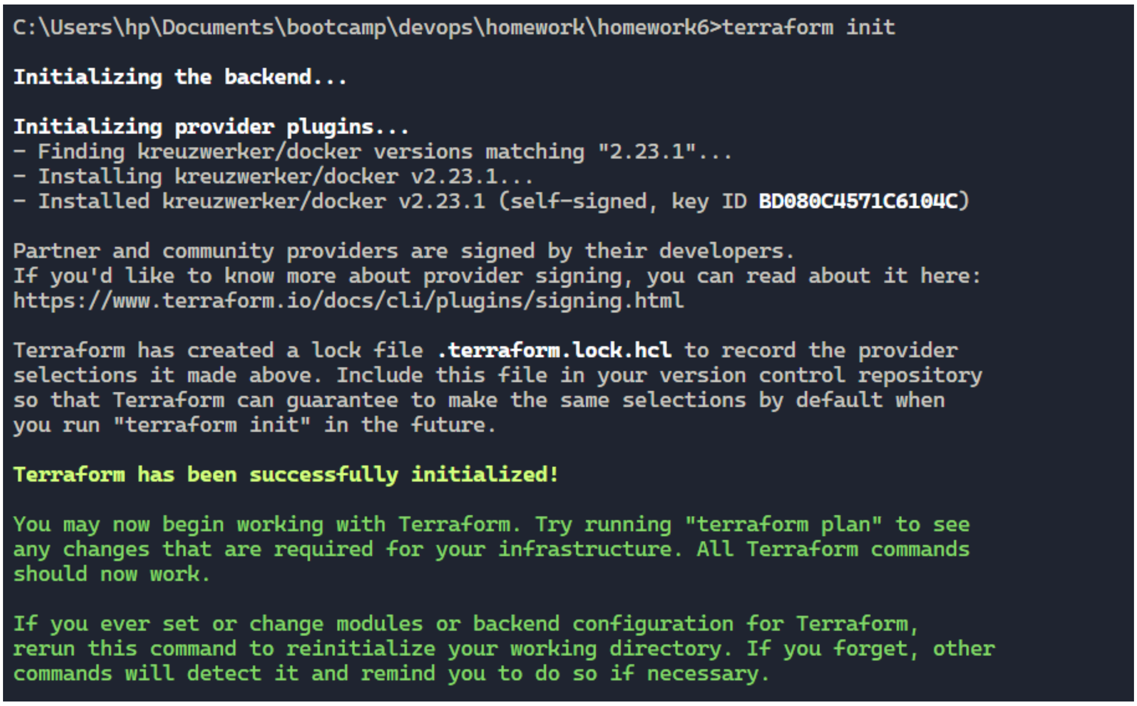
```
C:\Users\hp\Documents\bootcamp\devops\homework\homework6>terraform --version
Terraform v1.3.5
on windows_amd64
+ provider registry.terraform.io/kreuzwerker/docker v2.23.1
```

main.tf



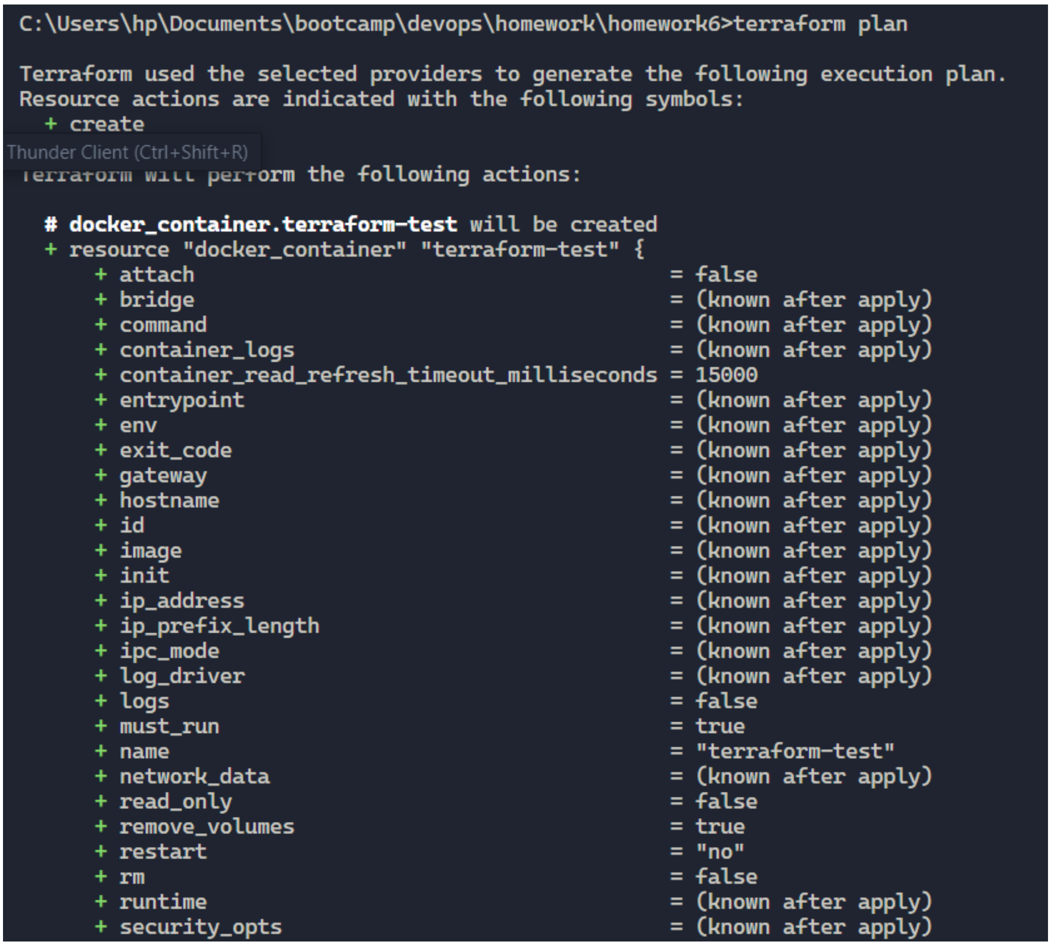
```
terraform {
  required_providers {
    docker = {
      source  = "kreuzwerker/docker"
      version = "2.23.1"
    }
  }
}

provider "docker" {
  host = "npipe:////.//pipe//docker_engine"
}

# Pulls the image
resource "docker_image" "ubuntu" {
  name = "ubuntu:latest"
}

# Create a container
resource "docker_container" "foo" {
  image = docker_image.ubuntu.image_id
  name  = "terraform-test"
  must_run = true
  command = ["sleep", "600"]
}
```

Terraform init

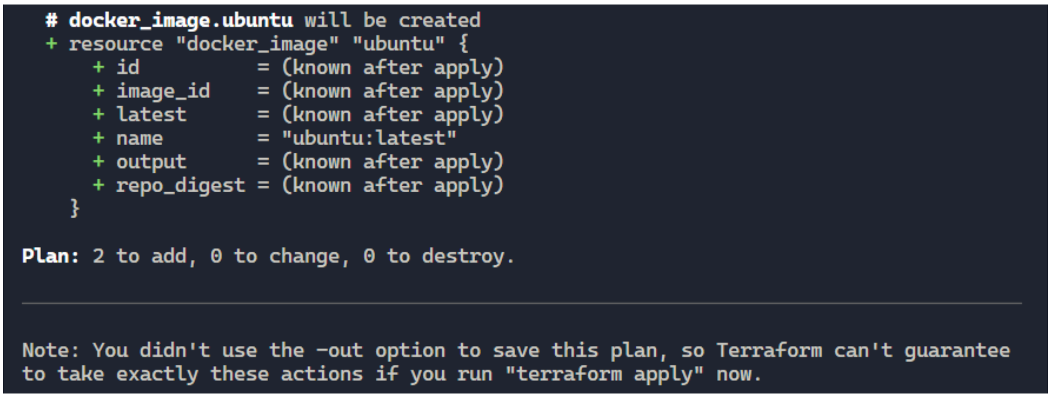


```
C:\Users\hp\Documents\bootcamp\devops\homework\homework6>terraform init

Initializing the backend...

Initializing provider plugins...
- Finding kreuzwerker/docker versions matching "2.23.1"...
- Installing kreuzwerker/docker v2.23.1...
- Installed kreuzwerker/docker v2.23.1 (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!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

Terraform plan

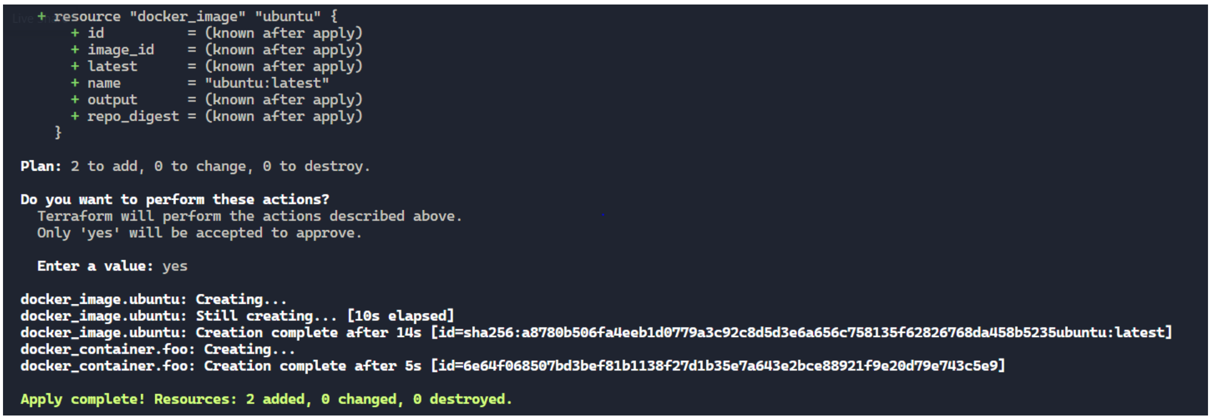


```
C:\Users\hp\Documents\bootcamp\devops\homework\homework6>terraform plan

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.terraform-test will be created
  + resource "docker_container" "terraform-test" {
      + attach                                      = false
      + bridge                                      = (known after apply)
      + command                                     = (known after apply)
      + container_logs                              = (known after apply)
      + container_read_refresh_timeout_milliseconds = 15000
      + 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)
      + ip_prefix_length                            = (known after apply)
      + ipc_mode                                    = (known after apply)
      + log_driver                                  = (known after apply)
      + logs                                        = false
      + must_run                                    = true
      + name                                        = "terraform-test"
      + network_data                                = (known after apply)
      + read_only                                   = false
      + remove_volumes                              = true
      + restart                                     = "no"
      + rm                                          = false
      + runtime                                     = (known after apply)
      + security_opts                               = (known after apply)
```



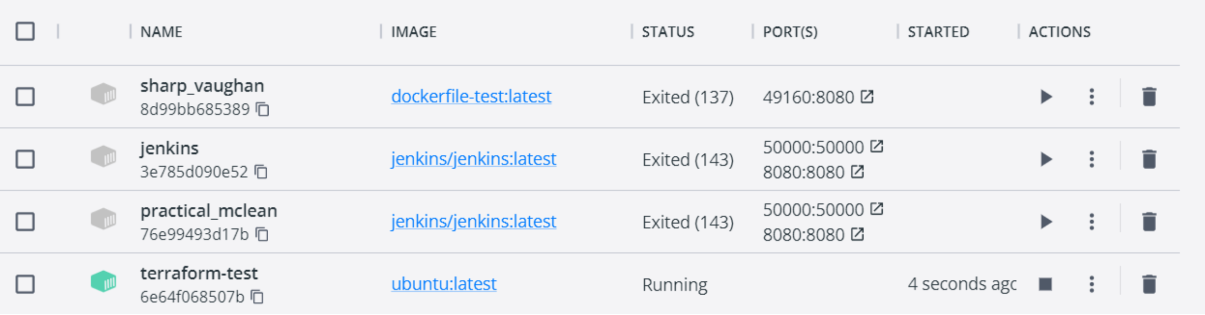
```
  # docker_image.ubuntu will be created
  + resource "docker_image" "ubuntu" {
      + id           = (known after apply)
      + image_id     = (known after apply)
      + latest       = (known after apply)
      + name         = "ubuntu:latest"
      + output       = (known after apply)
      +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

Terraform apply



```
  + 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.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

docker_image.ubuntu: Creating...
docker_image.ubuntu: Still creating... [10s elapsed]
docker_image.ubuntu: Creation complete after 14s [id=sha256:a8780b506fa4eeb1d0779a3c92c8d5d3e6a656c758135f62826768da458b5235ubuntu:latest]
docker_container.foo: Creating...
docker_container.foo: Creation complete after 5s [id=6e64f068507bd3bef81b1138f27d1b35e7a643e2bce88921f9e20d79e743c5e9]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```

Container running



□		NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
□		sharp_vaughan 8d99bb685389	dockerfile-test:latest	Exited (137)	49160:8080		▶ ⋮ 🗑
□		jenkins 3e785d090e52	jenkins/jenkins:latest	Exited (143)	50000:50000 8080:8080		▶ ⋮ 🗑
□		practical_mclean 76e99493d17b	jenkins/jenkins:latest	Exited (143)	50000:50000 8080:8080		▶ ⋮ 🗑
□		terraform-test 6e64f068507b	ubuntu:latest	Running		4 seconds agc	■ ⋮ 🗑