Download and install terraform

```
bryan@Bryan:~/terraform-homework$ sudo apt-get update && sudo apt-get install terraform
Hit:1 https://apt.releases.hashicorp.com focal InRelease
Hit:2 http://buaya.klas.or.id/ubuntu bionic InRelease
Hit:3 http://buaya.klas.or.id/ubuntu bionic-updates InRelease
Hit:4 http://buaya.klas.or.id/ubuntu bionic-backports InRelease
Hit:5 http://buaya.klas.or.id/ubuntu bionic-security InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  terraform
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 19.3 MB of archives.
After this operation, 64.8 MB of additional disk space will be used.
Get:1 https://apt.releases.hashicorp.com focal/main amd64 terraform amd64 1.1.9 [19.3 MB]
Fetched 19.3 MB in 16s (1209 kB/s)
Selecting previously unselected package terraform.
(Reading database ... 45526 files and directories currently installed.)
Preparing to unpack .../terraform_1.1.9_amd64.deb ...
Unpacking terraform (1.1.9) ...
Setting up terraform (1.1.9) ...
```

Then créate the file

```
terraform {
  required_providers {
    docker = {
      source  = "kreuzwerker/docker"
      version = "2.16.0"
    }
  }
}

provider "docker" {
}

# Pulls the image
resource "docker_image" "nginx" {
  name         = "nginx:latest"
  keep_locally = true
}

# Create a container
resource "docker_container" "nginx-server" {
  image = docker_image.nginx.latest
  name  = "nginx-server"
  ports {
     internal = 80
  }

  volumes {
    container_path = "/usr/share/nginx/html"
    host_path      =  "/data/"
    read_only      =  true
  }

}
```

Then launch with terraform init

```
bryan@Bryan:~/terraform-homework$ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding kreuzwerker/docker versions matching "2.16.0"...
- Installing kreuzwerker/docker v2.16.0...
- Installed kreuzwerker/docker v2.16.0 (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.
bryan@Bryan:~/terraform-homework$
```

Then run terraform plan

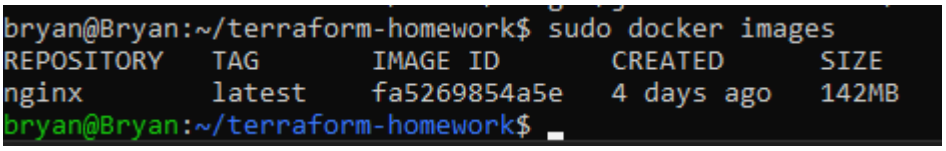
```
bryan@Bryan:~/terraform-homework$ sudo terraform plan

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-server will be created
  + resource "docker_container" "nginx-server" {
      + 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)
      + ip_prefix_length = (known after apply)
      + ipc_mode         = (known after apply)
      + log_driver       = (known after apply)
      + logs             = false
      + must_run         = true
      + name             = "nginx-server"
      + network_data     = (known after apply)
```

Now run: terraform apply

```
bryan@Bryan:~/terraform-homework$ sudo 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-server will be created
  + resource "docker_container" "nginx-server" {
      + 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)
      + ip_prefix_length = (known after apply)
      + ipc_mode         = (known after apply)
      + log_driver       = (known after apply)
      + logs             = false
      + must_run         = true
      + name             = "nginx-server"
      + network_data     = (known after apply)
```

```
docker_image.nginx: Creating...
docker_image.nginx: Still creating... [10s elapsed]
docker_image.nginx: Still creating... [20s elapsed]
docker_image.nginx: Still creating... [30s elapsed]
docker_image.nginx: Still creating... [40s elapsed]
docker_image.nginx: Creation complete after 40s [id=sha256:fa5269854a5e615e51a72b17ad3fd1e01268f278a6684c8ed3c5f0cdce3f2
30bnginx:latest]
docker_container.nginx-server: Creating...
docker_container.nginx-server: Creation complete after 1s [id=a9f2867e90da2501a2b240f9b4a85b9cb772f395c84f595581137f51d7
76a478]
```

And the last thing to do is check the image from Docker


bryan@Bryan:~/terraform-homework$ sudo docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
nginx latest fa5269854a5e 4 days ago 142MB
bryan@Bryan:~/terraform-homework$