NOTE: I'm using wsl, so all commands are coming from there.

Show the created directory, and the repository:

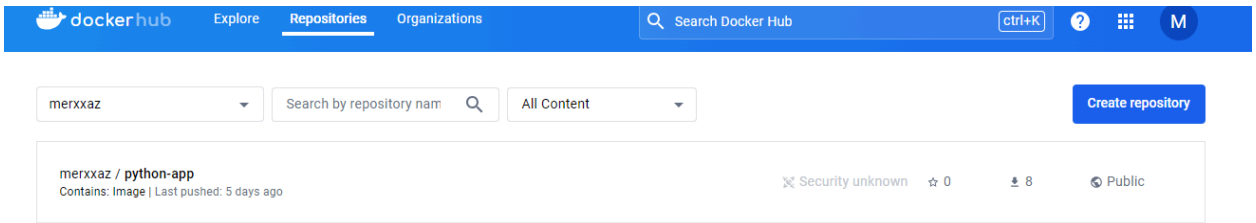
```
WSL at  miniProjects  7ms                                   bash  65.84%  12:11:26
>> ls
1stWeek  3rdWeek          4thWeek      bullshit     copy_copy3rdWeek
2ndWeek  3rdWeekFastApi   DataJenkins  copy3rdWeek  proof
WSL at  miniProjects  6ms                                   bash  65.75%  12:11:27
>> mkdir 5thWeek
WSL at  miniProjects  9ms                                   bash  65.71%  12:11:39
>> cd 5thWeek/
WSL at  5thWeek  4ms                                        bash  65.6%  12:11:45
```

```
WSL at  5thWeek  main ≠ ?1  9ms                             bash  65.55%  12:16:09
>> git add . && git commit -m "Brayan(W5): Creating the repo for the miniproject of 5th week of the
training of DevOps ..." && git remote add origin http://localhost:3000/BrayanMarin/Week5Terraform.git &
& git push -u origin main
[main (root-commit) cb7259e] Brayan(W5): Creating the repo for the miniproject of 5th week of the trai
ning of DevOps ...
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 hola.txt
Username for 'http://localhost:3000': BrayanMarin
Password for 'http://BrayanMarin@localhost:3000':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 270 bytes | 270.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote: . Processing 1 references
remote: Processed 1 references in total
To http://localhost:3000/BrayanMarin/Week5Terraform.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
```

Creating a folder to manage the terraform files, and showing which version of terraform I'm using

```
WSL at 5thWeek main ≡ -1 11ms bash 65.74% 12:19:04
mkdir terraform
WSL at 5thWeek main ≡ -1 6ms bash 65.91% 12:19:10
cd terraform/
WSL at terraform main ≡ -1 6ms bash 65.76% 12:19:13
terraform -version
Terraform v1.7.4
on linux_amd64
WSL at terraform main ≡ -1 336ms bash 65.75% 12:19:18
touch deployment.tf provider.tf service.tf
WSL at terraform main ≡ ?1 -1 5ms bash 65.77% 12:20:41
ls
deployment.tf  provider.tf  service.tf
WSL at terraform main ≡ ?1 -1 6ms bash 65.82% 12:20:42
```

After type anything on .tf files, I show the image of my python App uploaded in DockerHub:



Then, I created three files with the extension of terraform (.tf) to set up the project. The files are: deployment, services and provider:

```
You, 21 hours ago | 1 author (You)
resource "kubernetes_deployment" "pythonApp_API" {
  You, 21 hours ago | 1 author (You)
  metadata {
    name = "pythonapp-fastapi-deployment"
  }
  You, 21 hours ago | 1 author (You)
  spec {
    replicas = 2
    You, 21 hours ago | 1 author (You)
    selector {
      You, 21 hours ago | 1 author (You)
      match_labels = {
        app = "pythonAPP_FastAPI"
      }
    }
    You, 21 hours ago | 1 author (You)
    template {
      You, 21 hours ago | 1 author (You)
      metadata {
        You, 21 hours ago | 1 author (You)
        labels = {
          app = "pythonAPP_FastAPI"
        }
      }
      You, 21 hours ago | 1 author (You)
      spec {
        You, 21 hours ago | 1 author (You)
        container {
          name              = "pythonapp-fastapi-container"
          image             = "merxxaz/python-app:v1.0.0"
          image_pull_policy = "Always"
          You, 21 hours ago | 1 author (You)
          port {
            container_port = 8000
          }
          You, 21 hours ago | 1 author (You)
          resources {
            You, 21 hours ago | 1 author (You)
            limits = {
              cpu    = "500m" # 0.5 CPU
              memory = "512Mi"
            }
            You, 21 hours ago | 1 author (You)
            requests = {
              cpu    = "250m"
              memory = "50Mi"       You, 3 hours ago • Brayan(W5): A
            }
          }
        }
      }
    }
  }
}
```

```
You, 21 hours ago | 1 author (You)
resource "kubernetes_service" "pythonApp_API" {
  You, 21 hours ago | 1 author (You)
  metadata {
    name = "pythonapp-fastapi-service"
  }
  You, 21 hours ago | 1 author (You)
  spec {
    You, 21 hours ago | 1 author (You)
    selector = {
      app = "pythonAPP_FastAPI"
    }
    type = "NodePort"
    You, 21 hours ago | 1 author (You)
    port {
      port        = 8000
      target_port = 8000
      node_port   = 30007
    }
  }
}

```

```
You, 21 hours ago | 1 author (You)
terraform {
  You, 21 hours ago | 1 author (You)
  required_providers {
    You, 21 hours ago | 1 author (You)
    kubernetes = {
      source = "hashicorp/kubernetes"
    }
  }
}

You, 21 hours ago | 1 author (You)
provider "kubernetes" {
  config_path = "~/.kube/config"
}

You, 21 hours ago | 1 author (You)
terraform {
  You, 21 hours ago | 1 author (You)
  backend "local" {
    path = "terraform.tfstate"
  }
}        You, 21 hours ago • Brayan(W5): Addi
```

After that, I initialized the project with the following command:

```
WSL at  terraform  main ≡ ?2 -1  8ms  bash  59.14%  10:55:42

>> terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/kubernetes...
- Installing hashicorp/kubernetes v2.27.0...
- Installed hashicorp/kubernetes v2.27.0 (signed by HashiCorp)

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.
```

Then I did a terraform plan to see how gonna execute it the project:

```
WSL at  terraform  main ≡ ?2 -1  1s 752ms  bash  59.12%  10:55:46
>> 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:

  # kubernetes_deployment.pythonApp_API will be created
  + resource "kubernetes_deployment" "pythonApp_API" {
      + id               = (known after apply)
      + wait_for_rollout = true

      + metadata {
          + generation       = (known after apply)
          + name             = "pythonapp-fastapi-deployment"
          + namespace        = "default"
          + resource_version = (known after apply)
          + uid              = (known after apply)
        }

      + spec {
          + min_ready_seconds         = 0
          + paused                    = false
          + progress_deadline_seconds = 600
          + replicas                  = "2"
          + revision_history_limit    = 10

          + selector {
              + match_labels = {
                  + "app" = "pythonAPP_FastAPI"
                }
            }

          + template {
              + metadata {
                  + generation       = (known after apply)
                  + labels           = {
                      + "app" = "pythonAPP_FastAPI"
                    }
                  + name             = (known after apply)
                  + resource_version = (known after apply)
                  + uid              = (known after apply)
                }
              + spec {
                  + automount_service_account_token  = true
                  + dns_policy                       = "ClusterFirst"
                  + enable_service_links             = true
                  + host_ipc                         = false
```

And finally, I did an apply to apply the project instead

```
WSL at  terraform  main ≡ ?2 -1  691ms  bash  57.18%  10:56:35
>> 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:

  # kubernetes_deployment.pythonApp_API will be created
  + resource "kubernetes_deployment" "pythonApp_API" {
      + id                = (known after apply)
      + wait_for_rollout = true
```

```
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

kubernetes_service.pythonApp_API: Creating ...
kubernetes_deployment.pythonApp_API: Creating ...
kubernetes_service.pythonApp_API: Creation complete after 0s [id=default/pythonapp-fastapi-service]
kubernetes_deployment.pythonApp_API: Creation complete after 8s [id=default/pythonapp-fastapi-deployme
nt]

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

Then, before try in the Jenkins server, I tried locally to see that all was working correctly.

```
WSL at  terraform  main ≡ ?2 -1  16s 108ms  bash  58.54%  10:57:46
>> kubectl get po
NAME                                    READY   STATUS             RESTARTS          AGE
hello-javamaven-56fbf76cc4-hvq65        0/1     Completed          183 (5m8s ago)    5d16h
hello-javamaven-56fbf76cc4-mss8f        0/1     CrashLoopBackOff   182 (16s ago)     5d16h
hello-onlyjava-74ff67c65-2q646          0/1     Completed          205 (5m18s ago)   5d18h
hello-onlyjava-74ff67c65-q76f8          0/1     CrashLoopBackOff   204 (4m50s ago)   5d18h
python-app-service-645c98855f-8m9jd     1/1     Running            0                 23h
python-app-service-645c98855f-xpjd5     1/1     Running            0                 23h
pythonapp-fastapi-deployment-7969966c6-5pjrn   1/1   Running      0                 59s
pythonapp-fastapi-deployment-7969966c6-pz2mw   1/1   Running      0                 59s
WSL at  terraform  main ≡ ?2 -1  796ms  bash  58.9%  10:58:38
>> kubectl port-forward pythonapp-fastapi-deployment-7969966c6-5pjrn 8000:8000
Forwarding from 127.0.0.1:8000 -> 8000
Forwarding from [::1]:8000 -> 8000
```

Proof of testing the api locally

localhost:8000/calculate_cos/89

GLOBANT LINKS

```
{
    "cos": 0.5101770449416689
}
```

Now uploading the remaining files to gitea repository, to then try to build the Jenkins Pipeline:

Jenkinsfile Jenkinsfile

```
def remote = [:]
remote.name = 'brayanmarin'
remote.host = '172.28.92.49'
remote.allowAnyHosts = true

pipeline {
    agent any // 'agent' should be outside the 'environment' block and directly under 'pipeline'
    environment {
        dockerImageName = 'merxxaz/python-app:v1.0.0'
        dockerImage = ''
        SSH_CREDS = credentials('23f6de3f-0ddf-4ca8-9c0d-18960773cd0e')
        TERRAFORM_DIR = '/home/brayanmarin/DevOps/miniProjects/5thWeek/testFilesTerraform'
    }
    stages {
        stage('SSH Connection') {
            steps {      You, 21 hours ago • Brayan(W5): Adding Jenkinsfile and files of ter…
                script {
                    // Assuming SSH_CREDS_USR and SSH_CREDS_PSW are environment variables, they should be accessed as follows:
                    remote.user = env.SSH_CREDS_USR
                    remote.password = env.SSH_CREDS_PSW
                }
            }
        }
        stage('Checkout Source') {
            steps {
                sh 'ls'
            }
        }
        stage('Sending terraform files') {
            steps {
                script {
                    def terraformFiles = sh(script: 'find . -name \"*.tf\"', returnStdout: true).trim()
                    echo "Terraform files: ${terraformFiles}"

                    terraformFiles.tokenize('\n').each {
                        file ->
                            def filePath = file.replaceFirst('^\\./', '')
                            sshPut(remote: remote, from: filePath, into: TERRAFORM_DIR) // 'sshPut' should be a valid step, provided by an SSH Pipeline Steps plugin
                    }
                }
            }
        }
        stage('Terraform Init and Apply') { // Combined Terraform Init and Apply stages for simplicity
            steps {
                script {
                    sshCommand(remote: remote, command: "cd ${TERRAFORM_DIR} && terraform init && terraform apply -auto-approve") // '-auto-approve' is the correct flag, not '-auto'
                }
            }
```

```
WSL at  5thWeek  main ≡ ?2 -1  6ms   bash  58.91%  11:03:30
>> git status -s
 D hola.txt
?? Jenkinsfile
?? terraform/
```

Dashboard > Week5Terraform_Kubernetes >

- Configure
- Delete Pipeline
- Full Stage View
- Favorite
- Open Blue Ocean
- Rename
- Pipeline Syntax

Build History trend

Filter... /

- #6 Mar 13, 2024, 4:30 PM
- #5 Mar 13, 2024, 4:26 PM
- #4 Mar 13, 2024, 4:21 PM
- #3 Mar 13, 2024, 4:19 PM
- #2 Mar 13, 2024, 4:09 PM
- #1 Mar 13, 2024, 4:07 PM

Stage View

	Declarative: Checkout SCM	SSH Connection	Checkout Source	Sending terraform files	Terraform Init and Apply
Average stage times: (Average full run time: ~17s)	1s	288ms	378ms	1s	8s
#6 Mar 13 11:30 — 1 commit	260ms	388ms	379ms	1s	12s
#5 Mar 13 11:26 — 1 commit	198ms	209ms	365ms	1s	11s failed
#4 Mar 13 11:21 — 1 commit	4s	186ms	362ms	3s	2s failed
#3 Mar 13 11:19 — 2 commits	3s	275ms	429ms	1s	11s failed
#2 Mar 13 11:09 — 1 commit	1s	384ms	358ms	1s	3s failed

Permalinks

- Last build (#5), 3 min 45 sec ago
- Last failed build (#5), 3 min 45 sec ago

<u>Once in the cluster of Kubernetes, I use the address that kubernetes gives me (Kubernetes sends two, once with a tunnel, and the other with the IP of the cluster):</u>

```
WSL at  7ms                                                bash  59.66%  11:00:40
>> minikube ip
192.168.49.2
WSL at  330ms                                              bash  59.64%  11:00:55
>> curl http://192.168.49.2/30001/calculate_cos/89
curl: (7) Failed to connect to 192.168.49.2 port 80 after 1 ms: Connection refused
WSL at  77ms                                               bash  59.8%   11:01:30
>> curl http://192.168.49.2/30001/
curl: (7) Failed to connect to 192.168.49.2 port 80 after 0 ms: Connection refused
WSL at  12ms                                               bash  59.75%  11:01:38
>> curl http://192.168.49.2:30001/calculate_cos/89
{"cos":0.5101770449416689}  WSL at  24ms                                  bas
h  59.73%  11:02:14
>>
```

```
127.0.0.1:42031/calculate_sin/80
GLOBANT LINKS
1 {
2     "sin": -0.9938886539233752
3 }
```