

TP5 - Kubernetes

Installation:

Installation via Chocolatey (powershell)

```
choco install minikube -y  
minikube version
```

Exercise 1:

- Démarrer un cluster minikube en utilisant Docker comme driver :

```
minikube start --driver=docker --cpus=2 --memory=4096
```

- Vérifier que le cluster est accessible :

```
kubectl cluster-info  
kubectl get nodes
```

- Le node principal doit apparaître avec le statut Ready.
- Vérifier les pods systèmes par défaut :

```
kubectl get pods -A
```

- Vérifier les pods systèmes par défaut :

```
kubectl get pods -A
```

- Les pods dans les namespaces kube-system et kube-public doivent être en statut Running ou Completed.

- Création d'un namespace de travail
- Créer un fichier namespace-mlops-dev.yaml :

```
apiVersion: v1  
kind: Namespace  
metadata:  
  name: mlops-dev
```

- Appliquer le fichier :

```
kubectl apply -f namespace-mlops-dev.yaml
```

- Vérifier la création :

```
kubectl get namespaces
```

- Créer un fichier pod-nginx.yaml :

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-nginx
  labels:
    app: nginx-demo
spec:
  containers:
    - name: nginx
      image: nginx:1.27
      ports:
        - containerPort: 80
```

- Appliquer le Pod dans le namespace mlops-dev :
`kubectl apply -f pod-nginx.yaml -n mlops-dev`
- Vérifier l'état du Pod :

```
kubectl get pods -n mlops-dev
kubectl get pods -n mlops-dev -o wide
```

- Si le Pod n'est pas Running, inspecter les détails :
`kubectl describe pod pod-nginx -n mlops-dev`
- Creer un fichier deploiment deployment-nginx.yaml :

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: deploy-nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-demo
  template:
    metadata:
      labels:
        app: nginx-demo
    spec:
      containers:
        - name: nginx
```

```
image: nginx:1.27
ports:
  - containerPort: 80
```

- Appliquer le déploiement:

```
kubectl apply -f deployment-nginx.yaml -n mlops-dev
```

- Vérifier le déploiement passe:

```
kubectl get deployments -n mlops-dev
kubectl get pods -n mlops-dev -o wide
```

- Supprimer un pod

```
kubectl delete pod <nom-d-un-pod> -n mlops-dev
```

- Vérifier que le pod est recréé automatiquement:

```
kubectl get pods -n mlops-dev
```