

**Mise en situation professionnelle**

	Cours : Kubernetes
Sujet : Premiers pas	Numéro : 3
	Version : 1.0

**Objectifs :**

Préparer votre poste de travail

**Prérequis :**

aucun

**Principales tâches à réaliser :**

3 Faire connaissance avec Kubernetes.....	2
3.1 Cloner le projet localement.....	2
3.2 Construire l'image.....	3
3.3 Lancer l'image de deux façon différentes.....	3
a Lancer l'image via Docker.....	3
b Lancer avec kubectl.....	3

## 3 Faire connaissance avec Kubernetes

Nous allons maintenant démarrer un projet mais basé sur une image buildée sur le serveur docker. Ceci permet d'illustrer deux choses :

- Un serveur Kubernetes est avant tout un serveur docker
- En développement par exemple il est possible de 'Builder' une image sans passer par un registre externe.

### 3.1 Cloner le projet localement

Dans la machine locale, taper :

```
$ minikube ssh  
  
      _      _      _      _      _  
  _  _  _  _  _  _  _  _  _  _  _  
/ ' _ ` _ ` \ | / ' _ ` \ | | , < ( ) ( ) | ' _ ` \ / ' _ ` \  
| ( ) ( ) | | | | ( ) | | | | \ \ | ( ) | | | | ) ( _ _ /  
( _ ) ( _ ) ( _ ) ( _ ) ( _ ) ( _ ) \ _ _ / ' ( _ _ / ' \ _ _ )  
$ su -
```

Créer un répertoire pour construire nos images

```
# mkdir /data/docker  
# chown docker.docker /data/docker  
# exit  
$ cd /data/docker  
$ mkdir Build  
$ cd Build
```

## Récupérer le projet

```
$ git clone http://registry.formation.local:3000/snoel/Nginx.git
Cloning into 'Nginx'...
remote: Counting objects: 17, done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 17 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (17/17), done.
$
$ cd Nginx/
```

## 3.2 Construire l'image

```
$ docker build -t registry.formation.local:5000/snoel/nginx .
Sending build context to Docker daemon 94.72kB
Step 1/6 : FROM alpine:3.9
...
Successfully built 55ad0285d1f6
Successfully tagged registry.formation.local:5000/snoel/nginx:latest
$ docker push registry.formation.local:5000/snoel/nginx
```

## 3.3 Lancer l'image de deux façon différentes

### a Lancer l'image via Docker

```
# docker run -d -p 7000:80 registry.formation.local:5000/snoel/nginx
```

Accéder au projet avec votre navigateur, puis supprimer le projet.

```
$ docker stop <id>
$ docker rm $(docker ps -qa)
```

### b Lancer avec kubectl

Sortir de la machine Minikube, puis lancer avec kubectl

```
$ exit
logout
~$ kubectl create deployment nginx --image=registry.formation.local:5000/snoel/nginx
$ kubectl expose deployment nginx --type=NodePort --port=80
$ kubectl get services
NAME          TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
nginx         NodePort      10.103.72.216   <none>           80:31769/TCP     13s
$ kubectl cluster-info
Kubernetes master is running at https://192.168.99.100:8443
```

On accède à l'application sur :

<http://192.168.99.100:31769>