🧪 Module 1 Lab – Kubernetes Tooling Setup & Environment Check

This lab will help you install the required CLI tools, create a local Kubernetes cluster using `kind`, and verify that everything is set up properly.

🌟 Objectives

- Install and verify the following tools:
  - `kubectl`
  - `kind`
  - `helm`
- Create a local Kubernetes cluster with **Kubernetes v1.32**
- Use basic `kubectl` commands to explore the API

⚙️ Prerequisites

Ensure you have:

- Docker installed and running on your machine
- Internet access to download binaries

🛠️ Step 1: Install Required Tools

✅ kubectl (Kubernetes CLI)

Edixos Labs
hello@edixos.com
+33 6 47 43 99 30

Linux / macOS

```
ARCH=$(uname -m)
if [[ "$ARCH" == "arm64" || "$ARCH" == "aarch64" ]]; then
  ARCH=arm64
else
  ARCH=amd64
fi

curl -LO "https://dl.k8s.io/release/v1.32.0/bin/$(uname | tr
'[:upper:]' '[:lower:]')/${ARCH}/kubectl"
chmod +x kubectl
sudo mv kubectl /usr/local/bin/
kubectl version --client
```

Windows (PowerShell)

```
Invoke-WebRequest -Uri "https://dl.k8s.io/release/v1.32.0/bi
n/windows/amd64/kubectl.exe" -OutFile "kubectl.exe"
```

Make sure to add the path to `kubectl.exe` to your system's `PATH` variable.

▶ ▶ 🪟 **How to Add kubectl.exe to PATH on Windows**

1. 📁 **Move** `kubectl.exe` **to a permanent folder**, for example:

   ```
   C:\kubetools\kubectl.exe
   ```

2. 🛍️ **Open Environment Variables:**
   - Press `Win + S`, search for `Environment Variables`
   - Open **"Edit the system environment variables"**
   - Click **"Environment Variables…"**

Edixos Labs
hello@edixos.com
+33 6 47 43 99 30

3. ➕ **Edit the `Path` variable:**
   - In **System variables** (or *User variables*), select `Path` and click **Edit**
   - Click **New** and add:

     ```
     C:\kubetools
     ```

   - Click **OK** to apply changes
4. ✅ **Verify installation:** Open a new terminal and run:

   ```
   kubectl version --client
   ```

✅ kind (Kubernetes in Docker) – v0.29.0

🐧 Linux / 🍎 macOS (Intel + Apple Silicon)

```
ARCH=$(uname -m)
if [[ "$ARCH" == "arm64" || "$ARCH" == "aarch64" ]]; then
  ARCH=arm64
else
  ARCH=amd64
fi

curl -Lo ./kind "https://kind.sigs.k8s.io/dl/v0.29.0/kind-
$(uname | tr '[:upper:]' '[:lower:]')-$ARCH"
chmod +x ./kind
sudo mv ./kind /usr/local/bin/kind
kind version
```

✅ helm (Kubernetes package manager)

Linux / macOS / Windows

Edixos Labs
hello@edixos.com
+33 6 47 43 99 30

```
curl -fsSL -o get_helm.sh https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3
chmod 700 get_helm.sh
./get_helm.sh
```

🪟 Windows (PowerShell)

```
Invoke-WebRequest -Uri https://kind.sigs.k8s.io/dl/v0.29.0/kind-windows-amd64 -OutFile kind.exe
```

Add `kind.exe` to your `PATH` to run it from any terminal.

▼ ▼ 🪟 **How to Add kind.exe to PATH on Windows**

1. 📁 **Move `kind.exe` to a permanent folder**, for example:

   ```
   C:\kubetools\kind.exe
   ```

2. 🧭 **Open Environment Variables:**
   - Press `Win + S`, search for `Environment Variables`
   - Open **"Edit the system environment variables"**
   - Click **"Environment Variables…"**
3. ➕ **Edit the `Path` variable:**
   - In **System variables** (or *User variables*), select `Path` and click **Edit**
   - Click **New** and add:

     ```
     C:\kubetools
     ```

Edixos Labs
hello@edixos.com
+33 6 47 43 99 30

- Click **OK** to apply changes

4. ✅ **Verify installation:** Open a new terminal and run:

```
kind version
```

🚀 Step 2: Create a Kind Cluster (Kubernetes v1.32)

Create a config file `kind-config.yaml`:

```
kind: Cluster
apiVersion: kind.x-k8s.io/v1alpha4
name: cka-lab
nodes:
  - role: control-plane
    image: kindest/node:v1.32.3
```

Then create the cluster:

```
kind create cluster --config kind-config.yaml
```

> 📝 *Note: If the `v1.32.3` image is not available, use the closest supported version (e.g. `v1.31.1`) and mention this during the training.*

🔍 Step 3: Verify the Cluster

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

Edixos Labs
hello@edixos.com
+33 6 47 43 99 30

📖 Step 4: Explore the Kubernetes API

```
kubectl api-resources
kubectl explain pod
kubectl explain deployment.spec.template
```

✅ Checklist

- ☐ All tools (`kubectl` , `kind` , `helm`) installed and working
- ☐ Kind cluster created with Kubernetes v1.32
- ☐ `kubectl` returns cluster info and node details
- ☐ Able to use `kubectl explain` and list API resources

💬 What's Next?

You're ready to begin real cluster interactions! In the next module, we'll start managing workloads, objects, and resources in your new Kubernetes cluster.

Edixos Labs
hello@edixos.com
+33 6 47 43 99 30