Hands-On Lab (HOL): Kubernetes Nodes & Pods – Basic Cluster Exploration

Author: Sandeep Kaushik
Topic: Kubernetes Nodes, Pods, CLI Exploration

Learning Objective

By the end of this HOL, learners will:

- Understand what Nodes and Pods are in Kubernetes
- Learn how to view cluster nodes
- Explore pods across namespaces
- Create pods using CLI
- Create pods using YAML (dry-run method)
- Understand pod scheduling and node mapping

Learning Outcome

After completing this lab, participants will be able to:

- Explain Node and Pod in simple terms
- View cluster infrastructure
- Monitor pod placement
- Use kubectl commands confidently
- Create pods using both imperative and declarative methods

Simple Explanation

What is a Node?

A **Node** is a machine (VM/Server) where containers actually run.

What is a Pod?

A **Pod** is the smallest deployable unit in Kubernetes that runs containers.

Simple Example:

Real Life	Kubernetes
Building	Cluster
Floor	Node
Room	Pod
Person	Container

Hands-On Lab (HOL)

Step 1: View Cluster Nodes

```
kubectl get nodes
```

Step 2: View Node Details

```
kubectl get nodes -o wide
```

Step 3: View Pods (Default Namespace)

```
kubectl get pods
```

Step 4: View Pods Across All Namespaces

```
kubectl get pods -A
```

Step 5: View Pods Across All Namespaces with Node Info

```
kubectl get pods -A -o wide
```

Step 6: Create Pod (Imperative Way)

```
kubectl run pod1 --image=nginx
```

Step 7: Verify Pod Creation

```
kubectl get pods
```

Step 8: View Pod with Node Mapping

```
kubectl get pods -o wide
```

Step 9: Describe Pod

```
kubectl describe pod pod1
```

Step 10: Check Containers via Docker

```
docker ps -a
```

Step 11: Generate Pod YAML (Dry-Run Mode)

```
kubectl run pod2 --image=nginx --dry-run=client -o yaml > pod.yaml
```

Step 12: Verify YAML File

```
ls
```

Step 13: Edit Pod YAML

```
vi pod.yaml
```

Step 14: Create Pod Using YAML (Declarative Way)

```
kubectl apply -f pod.yaml
```

Step 15: Verify Pods

```
kubectl get pods
```

Imperative vs Declarative

Type	Meaning
Imperative	Direct command execution (`kubectl run`)
Declarative	YAML based (`kubectl apply -f`)

Flow Understanding

Cluster → Node → Pod → Container

Summary

- Nodes are worker machines
- Pods run containers
- Pods are scheduled on nodes
- CLI and YAML both are valid creation methods
- `-A` = all namespaces
- `-o wide` = extended details

Interview One-Liners

- "A node is a worker machine where containers run."
- "A pod is the smallest deployable unit in Kubernetes."
- "kubectl run is imperative, kubectl apply is declarative."
