Exploring the Options

In this lesson, we'll look into different ways to set up a Kubernetes cluster and figure out how to set it up locally.

WE'LL COVER THE FOLLOWING

Different Ways to Set up a Cluster

One of the goals of this course is to limit the learning expense to a minimum. True to that spirit, we'll run a local Kubernetes cluster for as long as possible. At one point, we'll have to switch to a hosted, multi-node Kubernetes cluster. We'll do our best to postpone that for as long as possible without limiting your learning experience. For now, we'll create a **local Kubernetes cluster** on your machine.

There are quite a few ways to set up a local Kubernetes cluster:

- We could, for example, create a few nodes with Vagrant (a tool for building and managing virtual machine environments in a single workflow) and execute quite a few shell commands that would convert them into a Kubernetes cluster.
- We could go even further and create a VirtualBox image that would have all the required software pre-installed and use it to create Vagrant VMs.
- We could also use Ansible to run provisioning of those images as well as to execute all the commands required to join VMs into a cluster.

There are many other things that we could do, **but we won't**.

At this point, the idea is not to teach you all the intricacies of setting up a Kubernetes cluster. Instead, we want to get you up to speed as fast as possible

and let you experience Kubernetes without sidelining that experience with installation details.

If we were using Docker Swarm, we'd have Docker for Mac or Windows (or run it natively on Linux) and execute a single `docker swarm init` command. That's all that's needed to create a local Docker Swarm cluster. *Can we accomplish the same simplicity with Kubernetes?*

The answer is **Minikube**.

Using Minikube to Set up a Cluster

Minikube creates a single-node cluster inside a VM on your laptop. While that is not ideal, since we won't be able to demonstrate some of the features Kubernetes provides in a multi-node setup, it should be more than enough to explain most of the concepts behind Kubernetes.

Later on, we'll move into a more production-like environment and explore the features that cannot be demonstrated in Minikube.

A Note to Windows Users

Please run all the examples from *GitBash* (installed through *Git*). That way the commands you'll see throughout the course will be same as those that should be executed on *MacOS* or any *Linux* distribution. If you're using Hyper-V instead of VirtualBox, you may need to run the *GitBash* window as an Administrator.

Before we dive into Minikube installation, there are a few prerequisites we should set up. The first in line is `kubectl`.