



DEVOPS-101

# **Docker & Kubernetes Fundamentals**

# Docker & Kubernetes Fundamentals

## OVERVIEW/PREREQUISITES/COURSE OUTLINE

<b>Skill Level</b>	:	Beginner
<b>Suitable for</b>	:	Anyone with basic knowledge of Linux Command Line Interface (CLI) and some coding experience, wanting to learn about containers (Docker), and orchestration (Kubernetes)
<b>Duration</b>	:	3 Days

In this course, you will learn about the concept, features, use cases, and benefits of containers, and the difference between containers and virtual machines. We will introduce you to Docker, some of its more commonly used command line interface (CLI) commands, and how they work. This training will teach you how to build container images using Dockerfiles, and you will also learn about container registries and the functionalities they provide to help developers work more securely and productively.

Furthermore, this course discusses container orchestration and how it helps create and manage the lifecycle of complex container environments. You will gain working knowledge on the most popular container orchestration platform: Kubernetes.

This course will familiarize you with the key architectural components of Kubernetes, such as the control plane components and controllers. You will learn about the basics of Kubernetes objects, and how specific Kubernetes objects such as Pods and Deployments work. We will introduce you to a few basic kubectl commands, some key concepts such as scaling, rolling updates, ConfigMaps, Secrets, and service bindings, and explain the relative benefits and shortcomings of both imperative and declarative commands.

## LEARNING OUTCOMES

- Use the Docker client to build images and run containers
- Write a Dockerfile, build an image and run it as a container, and store the image in a registry.
- Use the kubectl CLI commands to create resources on a Kubernetes cluster.
- Deploy and update applications in a Kubernetes cluster.
- Configure probes to implement best-practice application deployment patterns
- Troubleshoot deployment issues.
- Understand when and how to use the various Kubernetes Service Types and Networking Controller to direct traffic as required for applications within a cluster.
- Employ best practices for namespacing and labelling of Kubernetes resources.

## PREREQUISITES (OR EQUIVALENT TRAINING/EXPERIENCE)

- DEVOPS-102 – Linux Fundamentals

# COURSE OUTLINE

## Container Basics

- Virtualization vs Containerization
- Installing Docker
- Docker Architecture (daemon, client, registries, images, containers)
- Getting started
  - Running containers
  - Connecting to running containers

## Dockerfile & Building Images

- Dockerfile instructions
- Multi-stage builds
- Image layers
- Environment variables
- Container (Docker) registries

## Container Networking & Storage

- Types of Docker networks
- Service discovery
- Storage (volume)

## Kubernetes Basics

- Container orchestration
- Kubernetes Architecture
  - Control Plane
  - Worker Nodes
- Install and Set up Kubernetes (Minikube or Docker Desktop)
- Getting started
- YAML manifest
- Pod and container
- Declarative vs Imperative approach

## Managing Applications

- Run stateless application using a Deployment
  - Writing a Deployment Spec: PodTemplate, Replicas, Selector
  - Built-in Strategies: RollingUpdate, Recreate
- Other strategies
  - Canary
  - Blue/Green
- Configure Liveness, Startup, and Readiness Probes
- Configure resource requirements and limits: CPU, RAM,

## Kubernetes Networking

- Networking
  - Within a Pod
- Service

- Pod to Pod
- Services to Pods
- External to Cluster
- Working with multiple clusters
- Types: ClusterIP, NodePort, LoadBalancer
- Ingress and Ingress Controller (Contour)
- Namespaces

### Additional Application Features

- Deploying stateful applications
  - StatefulSet
  - Storage: Volume, PersistentVolume, PersistentVolumeClaim
- Application configuration
  - ConfigMaps (as environment variables vs. as file on mounted volumes)
  - Secrets

### Additional Workloads & Security

- Jobs and DaemonSets
- Network policies
- RBAC (role-based access control)
  - ClusterRole and Role
  - ClusterRoleBinding and RoleBinding

### Software and Hardware Requirements

- Docker
- Kubernetes





## Enquiries



+63 2 5322 2307



[training-sales@orangeandbronze.com](mailto:training-sales@orangeandbronze.com)