

**Certified Kubernetes Application Developer (CKAD)**

**Course Number:** CLD-114
**Duration:** 1 day

**Overview**

This live, instructor-led Certified Kubernetes Application Developer (CKAD) training course teaches attendees Kubernetes concepts and how to manage a Kubernetes application. This class also prepares attendees to take and pass the CKAD certification exam.

**Prerequisites**

* Experience with containers, particularly Docker
* A basic understanding of Kubernetes concepts and API resources
* Familiarity with a Unix environment, bash commands, and a command line-based text editor (optimally vim) and YAML

**Materials**

All attendees receive a copy of the instructor’s presentation and related code.

**Software Needed on Each Student PC**

A machine with access to a Kubernetes cluster, either local or remote (The recommended setup is to install minikube and kubectl.)

**Objectives**

* Understand when and how to apply Kubernetes concepts to manage an application
* Master tips and tricks to pass the CKAD exam
* Understand the ins and outs of the kubectl command-line tool
* Demonstrate competency in performing the responsibilities of Kubernetes application developers
* Solve real-world Kubernetes problems in a hands-on command-line environment
* Navigate and solve questions during the CKAD exam

**Outline**

* Exam Details and Resources
	+ Exam objectives and curriculum
	+ Candidate skills and the exam environment
	+ Time management tips and tricks
	+ Additional resources
	+ Practice exams
	+ Your main learning objective
* Core Concepts
	+ Kubernetes object creation and management
	+ understanding pods
	+ inspecting and configuring pods
	+ Creating and inspecting a pod
* Configuration
	+ ConfigMaps
	+ Secrets
	+ Resource boundaries
	+ Declare service accounts
	+ Create a pod to use a ConfigMap
	+ Create a pod to use a secret
	+ Create SecurityContexts for a pod
	+ Define a pod’s resource boundaries
	+ Use a service account
* Multicontainer Pods
	+ Multiple containers in a pod
	+ Init containers
	+ Sidecar patterns
	+ Adapter patterns
	+ Patterns
	+ Create an init container
	+ Implement the adapter pattern
* Observability
	+ Readiness probes
	+ Liveness probes
	+ Debugging existing pods
	+ Define a pod’s readiness probe and liveness probe
	+ Fix a misconfigured pod
* Pod Design
	+ Labels
	+ Annotations
	+ Deployments
	+ Jobs
	+ CronJobs
	+ Define and query labels and annotations
	+ Perform rolling updates and scale a deployment
	+ Create a scheduled container operation
* Services and Networking
	+ Services
	+ Deployments
	+ Network policies
	+ Route traffic to pods inside and outside of a cluster
	+ Restrict access to and from a pod
* State Persistence
	+ Volumes
	+ PersistentVolumes
* Conclusion