

Docker Containerization Boot Camp

Code:	DOCK-CON-BC
Length:	3 days
URL:	View Online

Learn the skills to radically simplify application deployment, IT workflow, version control, production support and much more with the power of Docker containerization. In recent years, few IT tools have been as disruptive or innovative as Docker. Lightweight and fast, the open-source Docker engine provides an environment to run your code as well as an efficient workflow to get the code from developer machines to the test environment, and rapidly into production. Docker streamlines the deployment of your applications into isolated containers, allowing versatile new solutions across the entire IT value chain.

This Docker training workshop is ideal for developers and operations staff who want to containerize and improve reliability throughout the entire software development lifecycle. To get the most out of the course, you should be familiar with some Linux basics, including package management, basic networking, and simple shell scripting.

This 3-day Docker Containerization training class is a continuous hands-on workshop that teaches you how to use Docker and the workloads for which it is best suited. From your clean new Docker install on the first day, you will work through continuous real-world use cases to learn pragmatic, immediately useful skills. Our real-world format gets you up and running as quickly as possible with the technology, and focuses on making sure you understand how to best integrate Docker into your workflow for maximum productivity as soon as you return to work.

Skills Gained

- Rapidly deploy applications - By completely avoiding full machine and/or OS virtualization, containers vastly reduce application size and delivery time.
- Develop for nearly universal portability - By bundling all application dependencies into one container.
- Implement version control - Recursive reuse makes applications much more lightweight. Version control and rollback are easy.
- Enjoy extremely lightweight, minimal overhead - Docker images are small. Delivery and deployment of new application containers are as fast as it gets.
- Simplify maintenance - Much less overhead and fewer dependencies mean far less maintenance.

Who Can Benefit

This workshop is ideal for developers and operations staff who want to containerize and improve reliability throughout the entire software development life cycle.

- System and software architects
- Developers
- Testers and QA teams
- Release engineers
- IT operations staff

- Site reliability engineers
- DevOps practitioners
- DBAs and data engineering teams

Course Details

Course Outline

Part 1: Introduction

- What can you use Docker for?
- A logical segregation of duties
- The relationship between Docker and SOA
- How Docker fits into the development lifecycle
- How Docker ensures consistency from development through UAT and staging, and on to production
- Example use cases of Docker in the real world

Part 2: The components of Docker

- Underlying technology
- Docker client and server
- Filesystem images
- Registries
- Containers
- Networking

Part 3: Getting set up to start using Docker

- Getting set up on Windows
- Getting set up on the Mac
- Trying out our first container
- Getting set up for production on Linux
- Tweaking your production environment for best performance
- User interfaces for Docker management

Part 4: Container management

- Container naming
- Starting and stopping containers
- Attaching to a container
- Seeing what is happening in a container
- Running a process inside a container
- Daemonizing a container
- Automatic container restarts
- Deleting containers when we are finished with them

Part 5: Docker images and repositories

- Docker images explained
- How Docker images work
- Getting a list of images
- Searching for images on a repository
- Pulling an image
- Creating our own image
- Specify an image in a Dockerfile
- Building Dockerfile images
- Using the build cache for templating
- Viewing the image we have created
- Launching a container using our new image

Part 6: Registries

- What is the Docker hub?
- Pushing images to the Docker hub
- Running your own internal Docker registry
- Testing the internal registry

Part 7: A simple use case

- A single container static website
- Setting up a container running Nginx
- Launching our static site
- Updating our static site from git or bitbucket

Part 8: Continuous integration with Docker

- How Docker enables and supports CI
- Getting set up for Jenkins and Docker
- A basic Jenkins job
- Multi-configuration jobs
- Drone
- Shippable

Part 9: A more complex use case: Multi container application stacks

- A container for our NodeJS application
- A base image for our Redis containers
- Creating our Redis back-end cluster
- Capturing logs
- Managing containers

Part 10: Docker orchestration and service discovery

- Getting set up with Fig
- Configuring the fig.yml file
- How to use Fig
- Console
- Running a Console cluster

Part 11: Integrating with configuration management

- Managing your Docker hosts with Chef / Puppet / Ansible
- Building containers using configuration management tools
- Managing running containers with configuration management

Part 12: Docker and DevOps

- Enabling collaboration with Docker
- Using Docker to streamline workflow
- Using Docker's version control capabilities to enable experimentation and learning
- Docker's role in the overall IT value chain
- Creating value and quality with Docker
- Enabling smoother flow of work

Part 13: Course conclusion, open discussion, and Q&A

- Going back to work with a plan
- What was not covered in this class
- Q&A with the instructor
- Goodbyes

Schedule (as of 3)

Date	Location	
Jul 22, 2024 - Jul 24, 2024	Virtual	Enroll
Oct 7, 2024 - Oct 9, 2024	Virtual	Enroll

[Download Whitepaper: Accelerate Your Modernization Efforts with a Cloud-Native Strategy](#)
 Get Your Free Copy Now