Mindluster Platform

# About the course What is Docker Docker Containers Images etc

What is Docker? Wikipedia defines Docker as an open-source project that automates the deployment of software applications inside containers by providing an additional layer of abstraction and automation of OS-level virtualization on Linux. Wow! That's a mouthful. In simpler words, Docker is a tool that allows developers, sys-admins etc. to easily deploy their applications in a sandbox (called containers) to run on the host operating system i.e. Linux. The key benefit of Docker is that it allows users to package an application with all of its dependencies into a standardized unit for software development. Unlike virtual machines, containers do not have high overhead and hence enable more efficient usage of the underlying system and resources. What are containers? The industry standard today is to use Virtual Machines (VMs) to run software applications. VMs run applications inside a guest Operating System, which runs on virtual hardware powered by the serverâ€™s host OS. VMs are great at providing full process isolation for applications: there are very few ways a problem in the host operating system can affect the software running in the guest operating system, and vice-versa. But this isolation comes at great cost â€” the computational overhead spent virtualizing hardware for a guest OS to use is substantial. Containers take a different approach: by leveraging the low-level mechanics of the host operating system, containers provide most of the isolation of virtual machines at a fraction of the computing power. Why use containers? Containers offer a logical packaging mechanism in which applications can be abstracted from the environment in which they actually run. This decoupling allows container-based applications to be deployed easily and consistently, regardless of whether the target environment is a private data center, the public cloud, or even a developerâ€™s personal laptop. This gives developers the ability to create predictable environments that are isolated from the rest of the applications and can be run anywhere. From an operations standpoint, apart from portability containers also give more granular control over resources giving your infrastructure improved efficiency which can result in better utilization of your compute resources. Due to these benefits, containers (& Docker) have seen widespread adoption. Companies like Google, Facebook, Netflix and Salesforce leverage containers to make large engineering teams more productive and to improve utilization of compute resources. In fact, Google credited containers for eliminating the need for an entire data center. What will this tutorial teach me? This tutorial aims to be the one-stop shop for getting your hands dirty with Docker. Apart from demystifying the Docker landscape, it'll give you hands-on experience with building and deploying your own webapps on the Cloud. We'll be using Amazon Web Services to deploy a static website, and two dynamic webapps on EC2 using Elastic Beanstalk and Elastic Container Service. Even if you have no prior experience with deployments, this tutorial should be all you need to get started.

**Programming Category's Courses**

# Course Lesson(10)


Lesson 1 : **Docker Tutorial What is Docker Docker Containers Images etc**

Lesson 2 : **Docker Tutorial Docker Container Tutorial for Beginners**

Lesson 3 : **Docker Container Tutorial How to build a Docker Container Image**

Lesson 4 : **Complete Docker Course Real World Docker Complete Docker Tutorial Course**

Lesson 5 : **Why Docker is the Perfect Fit for Microservices Production and Local Advantages**

Lesson 6 : **Scaling Docker 1 High Availability Docker DevOps with CoreOS Etcd Fleet Flannel Nginx**

Lesson 7 :
**Scaling Docker 2 Tutorial on Scheduling Docker Containers across CoreOS machine with Fleet**

Lesson 8 : **Scaling Docker 3 Docker Service Registration with Etcd Flannel**

Lesson 9 : **Scaling Docker 4 Docker Service Discovery and Load Balancing**

Lesson 10 : **Scaling Docker 5 Routing DNS to Docker images running on a CoreOS Cluster**

# Related courses

**React Projects Workshops**

**Node js back end JavaScript**

**Python programming language**

**Social Network Theme UI With Sass**

**Full Stack React Django**

**Build a Node js App With Sequelize**



for Business Contact
business@mindluster.com