



**Hewlett Packard
Enterprise**

Docker HODS3S

HPE course number	HODS3S
Course length	3 days
Delivery mode	ILT, VILT
View schedule, local pricing, and register	View now
View related courses	View now

Why HPE Education Services?

- IDC MarketScape leader 4 years running for IT education and training*
- Recognized by IDC for leading with global coverage, unmatched technical expertise, and targeted education consulting services*
- Key partnerships with industry leaders OpenStack®, VMware®, Linux®, Microsoft®, ITIL, PMI, CSA, and (ISC)2
- Complete continuum of training delivery options—self-paced eLearning, custom education consulting, traditional classroom, video on-demand instruction, live virtual instructor-led with hands-on lab, dedicated onsite training
- Simplified purchase option with HPE Training Credits

Linux containers are changing the way companies think about service development and deployment. Containers play a vital role in the modern data-center, and Docker is leading the way. This course covers all the core features of Docker including: container creation and management, interacting with Docker hub, using Dockerfile to create and manage custom images, advanced Docker networking (how to safely expose container services to the world, and link containers), the use of Docker volumes to manage persistent data, and Docker Compose to build multi-container applications. Emphasis is placed on best practices and how to secure Docker installations and containers. The course culminates with comprehensive labs where students use Docker, Git, and a continuous integration server to automate the testing of containerized applications.

Prerequisites:

Proficiency with the Linux CLI. A broad understanding of Linux system administration.

Detailed Course Outline

Container Technology Overview	<ul style="list-style-type: none"> • Application Management Landscape • Application Isolation • Resource Measurement and Control • Container Security • Open Container Initiative 	<ul style="list-style-type: none"> • Docker Ecosystem • Lab Tasks <ul style="list-style-type: none"> ◦ Container Concepts runC ◦ Container Concepts Systemd
Installing Docker	<ul style="list-style-type: none"> • Docker Architecture • Starting the Docker Daemon • Docker Daemon Configuration • Docker Control Socket • Enabling TLS for Docker 	<ul style="list-style-type: none"> • Lab Tasks <ul style="list-style-type: none"> ◦ Installing Docker ◦ Protecting Docker with TLS ◦ Install Docker via Docker Machine
Managing Containers	<ul style="list-style-type: none"> • Creating a New Container • Listing Containers • Viewing Container Operational Details • Running Commands in an Existing Container • Interacting with a Running Container • Stopping, Starting, and Removing Containers 	<ul style="list-style-type: none"> • Copying files in/out of Containers • Inspecting and Updating Containers • Lab Tasks <ul style="list-style-type: none"> ◦ Managing Containers ◦ Configure a docker container to start at boot
Managing Images	<ul style="list-style-type: none"> • Docker Images • Listing and Removing Images • Searching for Images • Downloading Images • Committing Changes • Uploading Images 	<ul style="list-style-type: none"> • Export/Import Images • Save/Load Images • Lab Tasks <ul style="list-style-type: none"> ◦ Docker Images ◦ Docker Platform Images
Creating images with Dockerfile	<ul style="list-style-type: none"> • Dockerfile • Caching • docker build • Dockerfile Instructions • ENV and WORKDIR 	<ul style="list-style-type: none"> • Running Commands • Getting Files into the Image • Defining container executable best practices • Lab Tasks <ul style="list-style-type: none"> ◦ Dockerfile Fundamentals
Docker Networking	<ul style="list-style-type: none"> • Overview • Data-Link Layer Details • Network Layer Details • Hostnames and DNS • Local Host <--> Container • Container <--> Container (same node) • Container <--> Container: Links • Container <--> Container: private network 	<ul style="list-style-type: none"> • Managing private networks • Remote Host <--> Container • Multi-host networks with overlay driver • Lab Tasks <ul style="list-style-type: none"> ◦ Docker Networking ◦ Docker Ports and Links ◦ Multi-host networks

Detailed Course Outline

Docker Volumes

- Volume Concepts
 - Creating and Using Volumes
 - Managing volumes (cont.)
 - Changing Data in Volumes
 - Removing Volumes
 - Backing up Volumes
 - SELinux Considerations
 - Mapping Devices
 - Lab Tasks
 - Docker Volumes
-

Docker Compose / Swarm

- Concepts
 - Compose CLI
 - Defining a Service Set
 - Docker Swarm
 - Lab Tasks
 - Docker Compose
 - Docker Swarm
-

Docker Registry

- Lab Tasks
-

Continuous Integration with GitLab, GitLab CI, and Docker

- Lab Tasks
 - GitLab and GitLab CI Setup
 - Unit and Functional Tests
-

Learn more at hpe.com/ww/learndevops

Follow us:



© Copyright 2017 Hewlett Packard Enterprise Development LP. The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Microsoft is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries. The OpenStack Word Mark is either a registered trademark/service mark or trademark/service mark of the OpenStack Foundation, in the United States and other countries and is used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation or the OpenStack community. Pivotal and Cloud Foundry are trademarks and/or registered trademarks of Pivotal Software, Inc. in the United States and/or other countries. Linux is the registered trademark of Linus Torvalds in the U.S. and other countries. VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions.