

# Docker y Orquestación de contenedores con Docker Swarm y Kubernetes

Detalles del curso:

**Duración:** 20 horas

**Modalidad:** Videoconferencia

Información e inscripción:

685 457 610 - 615 844 193

[formacion@laberit.com](mailto:formacion@laberit.com)

**LĀBERIT**

CENTRO DE  
FORMACIÓN TIC.

## Introducción

Docker es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos.

Debido a su tamaño pequeño y su orientación a aplicación, los contenedores son ideales para entornos de entrega ágiles y arquitecturas basadas en microservicios. Sin embargo, en un entorno que usa contenedores y microservicios, se puede tener que hacer seguimiento de cientos o miles de componentes. Es posible que pueda administrar manualmente unas docenas de máquinas virtuales o servidores físicos, pero no existe forma de administrar correctamente un entorno de contenedores de escala de producción sin automatización. Esta tarea debe realizarla el orquestador, que es un proceso que automatiza y administra un gran número de contenedores y cómo interactúan entre sí.

## ¿A quién va dirigido?

Administradores, DevOps e Ingenieros de Sistemas que necesiten crear contenedores de Docker en su día a día profesional.

## Objetivos didácticos

- Aprenderás a instalar y configurar Docker para Linux y para Windows
- Usarás los comandos y herramientas más usados en este tipo de infraestructura
- Crearás contenedores de distinto tipo
- Crearás imágenes personalizadas para generar contenedores
- Manejarás Docker Hub y subirás tus imágenes a ese repositorio
- Usarás Docker Compose para orquestar servicios entre contenedores
- Veremos como se crea un cluster con Docker Swarm
- Aprenderás las bases de Kubernetes y como trabajar con este orquestador de contenedores para gestionar clusters
- Numerosas prácticas en cada sección para que puedas poner en marcha lo aprendido

## Requisitos previos

- Necesitas algunos conocimientos de Linux o de Windows para empezar
- Necesitas disponer de un PC con Linux o con Windows 10
- Acceso a Internet para realizar los ejemplos del curso. Este punto es obligatorio

## Programa

### 1. Introducción a Docker

- ¿Qué es Docker?
- Build, Ship & Run
- Docker Architecture & Workflow
- Docker Engine & Componentes
- Docker in deep
- Docker Ecosystem
- Docker en Clouds
- Instalación y configuración

### 2. Contenedores

- Introducción
- Máquinas Virtuales vs Contenedores
- Crear un container con run
- Ciclo de vida de un container
- Crear container con estado Created
- Logs de containers
- Inyectar comandos a un container
- Attach a container
- Limitando recursos (CPU & RAM)
- Docker stats
- Inspección de un container
- Eliminar un container
- Eliminar containers en estado Exited
- Containers with uid:gid for run & exec
- Crear image mediante commit
- Exportar container como image
- Recomendaciones

- Ejercicios

## 3. Docker Images

- Búsquedas y descargas
- Versionado mediante Tag
- Image mediante Dockerfile
- Containers as non-root
- Health checks
- CMD vs Entrypoint
- Multi-stage
- Testing de images
- Anatomía de un image
- Inspección de un image
- Historial de image
- Importar y exportar images
- Eliminar un image
- Eliminar images no usadas
- Docker Hub
- Crear Repositorio privado
- Recomendaciones
- Ejercicios

## 4. Storage y Persistencia de data

- Storage
- Tipos
  - Bind mounts
  - Volumes
  - tmpfs
- Casos de uso
- Inspección de volumes
- Eliminar un volume
- Eliminar volumes no usados
- Plugins & Storages
- Recomendaciones
- Ejercicios

## 5. Networking

- Forwarding de puertos
- Tipos: bridge, host or None
- Asignación de ip estática
- Inspección de network
- Eliminar network
- Eliminar networks no usadas
- Recomendaciones
- Ejercicios

## 6. Docker Compose

- Intro
- Orchestrating
- Services
- Run & exec
- Depends and wait
- Scaling
- Exportar docker-compose.yml a Kubernetes
- Recomendaciones
- Ejercicios

## 7. Docker Swarm

- Clustering
- Nodes: Manager & Worker
- Stacks – Services & Tasks
- Mode: Global – Replicated
- Placement
- Load balancing
- Port binding
- Labels
- Scaling
- Visualizer
- Secrets & configs
- Deploy desde un docker-compose.yml
- Recomendaciones
- Ejercicios

## 8. Introducción a Kubernetes

- Que es k8s?
- Componentes
- Objects & Manifests
- Labels, selectors, annotations
- Stateless vs Statefull
- Kubectl syntax
- Nodes
- Cluster info
- Dashboard

## 9. Creación de un Cluster

- Cloud: Aws, Google, Azure y Digital Ocean.
- K8s en AWS con Kops
- The hard way
- Local con Minikube

## 10. Pods

- Introducción
- Correr una aplicación
- Ejecutar un comando personalizado
- Explorar la aplicación
- Ejecutar comandos en un container
- Inspeccionar el container
- Logs del container
- Proxy del container
- Correr dos containers
- Enviroment Variables
- Recomendaciones
- Ejercicios

## 11. Deployment

- Replicas – Replicaset
- Inspect your Deployment
- Update
- Rolling update and Rolling back

- Limitando Recursos
- Monitoreo de recursos (heapster)
- Scaling and Autoscaling
- Recomendaciones
- Ejercicios

## 12. Configuraciones

- ConfigMaps
- Secrets
- Recomendaciones
- Ejercicios

## 13. Volumes

- EmptyDir
- Lifecycle
- Persisting Data
- Persisting Data Claim
- Config map as Volume
- Recomendaciones
- Ejercicios

## 14. Services

- ClusterIP (default internal)
- Node Port
- Load Balancer (external or internal)
- External Name
- Services Discovery
- Ingress
- Recomendaciones
- Ejercicios

**Acreditado:**

Microsoft **Imagine Academy**  
Program Member

**Fundación Estatal**  
PARA LA FORMACIÓN EN EL EMPLEO 

**LABORA**  
Servei Valencià d'Ocupació i Formació

**CertiProf® | Partner**

 **Pearson**  
**VUE**  
Authorised  
Test Centre

  
A PEARSON VUE BUSINESS  
**AUTHORIZED TESTING CENTER**