

Training Program: Introduction à la containerisation d'applications

Published on 07/04/2026

DESCRIPTION OF THE TRAINING:

This two-day course provides a comprehensive introduction to the fundamentals of containerization and modern practices for deploying applications in a containerized environment.

Participants will discover what containers are (origin and purpose) and their functionalities (images, volumes, networking), how to create containerized applications (dockerfile, image libraries, multi-container applications, etc.), the associated tools (Docker, Helm), their implementation on the 2 main operating systems (Linux and Windows), how to integrate them into a CI/CD chain (Github Actions or Azure DevOps).

Developers will learn how to containerize their development environment (Remote Development with Visual Studio or VSCode).

In conclusion, an introduction to container orchestration (with Docker 'swarm mode' and Azure ACI/ACA/AKS services) and the concept of GitOps.

EDUCATIONAL OBJECTIVES:

At the end of this training, participants will be able to:

- Run and configure containers (network, volumes, environment variables)
- Build a multi-container application with Docker Compose
- Automate the construction and deployment of images via a CI/CD chain
- Deploy a containerized application to a cloud environment (Azure)

TEACHING METHODS:

- This training will consist of theory and technical workshops that will allow you to be quickly operational.
- Material: A course material will be given to participants in electronic format.
- Evaluation: The achievements are evaluated throughout the training by the trainer (regular questions, practical work, MCQs or other methods).
- Satisfaction: at the end of the training, each participant answers an evaluation questionnaire which is then analysed with a view to maintaining and improving the quality of our training.
- Follow-up: a sign-in sheet is signed for the half-day by each of the participants.

TRAINING PROGRAM:

Siège administratif et adresse de correspondance : Cellenza - 156, Boulevard Haussmann – 75008 – Paris

Siège social : 16, place de l'Iris, Esplanade Business Centre-Tour CB21 – 92400 – Courbevoie. SAS au capital variable de 50.000€ minimum

RCS Nanterre 529 602 161 – N° de déclaration d'activité 11921845892

training@cellenza.com | Tél : +33 (0)1 45 63 14 29 | www.training.cellenza.com

Why containerize applications

- Know the added value of a cloud application
- Learn about "The 12 Factors" of a Successful Cloud Native Application
- Understand the differences between Linux vs. Windows vs. other operating system.

Know the containerization tools:

- Docker Engine / Docker Desktop
- Docker Hub / Registry d'image
- Development environment (VSCode or Visual Studio)
- Helm

Containerize a C#, Dotnet, or Node application

- Create a DockerFile and build images
- Debug a containerized application with VisualStudio or VSCode
- Create a multi-container application with Docker Compose (networking)

Managing data in a containerized context

- Manage storage volume for persistent data

Know the lifecycle and safety

- Use Image Versions and Tags
- Manage secrets and configuration data
- Apply best practices for versioning and securing images (hardened images)

Automate container image construction and deployment

- Use Github Actions or Azure DevOps

Beyond container creation:

- Learn about container orchestration with Docker 'swarm-Mode' and Azure container services
- Introduction to the concept of "GitOps"

PREREQUISITES:

To take this course, participants will need to have basic knowledge of system administration or development. They will also need to have a general understanding of how Git works as well as basic command line knowledge (Linux or equivalent)

DURATION: 2 days (14 hours)

TARGET:

- Developers who want to understand modern application packaging and deployment mechanisms
- Entry-level system administrators or DevOps
- Technical teams wishing to adopt a GitOps approach
- Technical project managers who want to understand the issues and concepts

LEVEL: Intermediate