

LI460 Jenkins

Kurzbeschreibung:

Teilnehmende lernen Installation und Betrieb von Jenkins sowie die Anbindung an gängige Entwicklungsumgebungen. Behandelt wird, wie Jenkins Continuous Integration und Delivery unterstützt, Tests automatisiert und Qualität sowie Prozesse verbessert. Praxisnahe Anwendungsfälle verdeutlichen den Nutzen von Jenkins in agilen Softwareentwicklungsprojekten.

Zielgruppe:

Das Training **LI460 Jenkins** richtet sich in erster Linie an System-Administratoren.

Voraussetzungen:

Teilnehmer des Workshops **LI460 Jenkins** sollten gute Kenntnisse der Linux-Systemadministration mit RHCE, SCA, LPIC1 oder vergleichbar mitbringen.

Folgende Trainings werden je nach Vorkenntnissen empfohlen:

- [LI100 Linux Basics](#)
- [NT100 Netzwerk Administration Basics](#)
- [PL100 Shell awk sed Basics](#)

Sonstiges:

Dauer: 2 Tage

Preis: 1150 Euro plus Mwst.

Ziele:

Der Kurs **LI460 Jenkins** verfolgt folgende Ziele:

- Sie installieren Jenkins und nehmen grundlegende Konfigurationen für den Betrieb vor
- Sie verbinden Jenkins mit gängigen Entwicklungsumgebungen und integrieren es in bestehende Workflows
- Sie erstellen und konfigurieren automatisierte Tests innerhalb von Jenkins
- Sie analysieren praxisnahe Anwendungsfälle und übertragen die Erkenntnisse auf eigene Projekte

Inhalte/Agenda:

- **♦ Einführung**
 - ♦ ♦ Was ist Continuous Integration Testing/Delivery, wo findet es Anwendung?
 - ♦ ♦ Übersicht über Lösungen für Continuous Integration (Jenkins, GitLab-CI, Travis, ...)
- **♦ Übersicht über Jenkins**
 - ♦ ♦ Jenkins-Übersicht (Geschichte, Fork)
 - ♦ ♦ Jenkins 2 als Single-Installation/-Konfiguration
 - ♦ ♦ Plugin-Übersicht
- **♦ Test-Jobs erstellen**
 - ♦ ♦ einfachen Job erstellen (Bash, Maven, Ant)
 - ♦ ♦ Pipeline – Verbinden von Jobs
 - ♦ ♦ Benachrichtigung bei Fehlern/Trends
- **♦ Tools verbinden**
 - ♦ ♦ Versionsverwaltungen: GitLab, GitHub, SVN
 - ♦ ♦ Überblick: Dynamische Worker mit Cloud-Anbindung oder Docker
- **♦ Ausblick**
 - ♦ ♦ Continuous Delivery mit Jenkins Anwendungen in Docker und push Registry
 - ♦ ♦ Ticketing am Beispiel von Jira