

Category: DevOps

GitLab CI/CD # 010101

Das Training läuft vom 20.04.2026 bis zum 24.04.2026 und es wird auf Englisch gehalten.

Es findet sich vom 08:30 bis 17:00 statt.

Das Training findet sich am Ort in Circle Business Center am Zürich Flughafen und per Online statt. (Hybrid)

Key Learnings

- Building CI/CD pipelines in GitLab
- Job Keywords
- Variables, artifacts & caching
- Pipeline types and triggers
- GitLab Runner – characteristics, types & configuration
- Build & deployment via Docker & Kubernetes
- Introduction and differentiation of Git workflows

Requirements

- Basic knowledge of Git and GitLab is nice to have

GitLab CI/CD Training	
Module	
<p>1. GitLab & Overview of CI/CD</p> <ul style="list-style-type: none"> • approaches and added value • First steps with GitLab • GitLab CI/CD architecture • GitLab AutoDevOps as inspiration for concrete pipelines • GitLab Runner and Executor 	
<p>2. GitLab Excersices</p> <ul style="list-style-type: none"> • Master GitLab from basics to advanced features • Set up and manage GitLab repositories efficiently • Implement effective branching and merging strategies 	
<p>3. CI/CD Jobs</p> <ul style="list-style-type: none"> • Jobs, stages and workflows • Variables and their sources/hierarchy • Dependencies and needs • Artifacts and cache • Tags, images, services • Triggers and schedules • Rules • Environments • Includes, templates, anchors and references 	
<p>4. CI/CD Jobs Exercises</p>	
<p>5. Debugging & Registries</p> <ul style="list-style-type: none"> • Debugging CI/CD configurations • Configure and optimize GitLab CI/CD Pipelines • Merge requests and feature branches • Utilize GitLab's Package Registry and Container Registry 	
<p>6. Project Exercises</p>	
<p>7. Kubernetes & Terraform</p> <ul style="list-style-type: none"> • Integrate GitLab with cloud services like AWS and Kubernetes (EKS) • Leverage GitLab's Terraform integration for infrastructure as code 	

<ul style="list-style-type: none"> • Implement automated testing and code quality checks with SonarQube 	
<p>8. Docker & Ansible</p> <ul style="list-style-type: none"> • Use GitLab with Docker and DockerHub for containerized workflows • Integrate GitLab with Ansible for configuration management • Manage Terraform states within GitLab 	
<p>9. Performance and Scalability</p> <ul style="list-style-type: none"> • Connect GitLab with Jira and Confluence for seamless project management • Implement security best practices in GitLab workflows • Administer and optimize GitLab for performance and scalability 	
<p>10. Other Tools</p> <ul style="list-style-type: none"> • Apply GitLab's features to real-world development scenarios • Collaborate effectively using GitLab's team-oriented tools • Set up Telegram notifications for GitLab events 	