

# GitLab CI/CD in Practice

## From Fundamentals to Pipelines - 1 day

FA-0234 · 1 day

DevOps, Cloud & Infrastructure

---

### Introduction

Learn how to implement and manage CI/CD with GitLab like a seasoned DevOps pro.

---

Continuous Integration and Continuous Deployment (CI/CD) are core practices in modern DevOps workflows that help teams deliver high-quality software faster and more reliably. *GitLab* integrates powerful CI/CD capabilities directly into its DevOps platform, allowing developers and operations teams to automate builds, tests, and deployments in a consistent way.

This 1-day course gives you hands-on, practical knowledge of how CI/CD works in GitLab — from core principles and pipeline basics to runners, jobs, and deployment automation. The content reflects industry-relevant skills used in real projects and taught by an instructor with deep experience in delivering production-ready CI/CD solutions.

---

### Learning outcomes

By the end of this course you will be able to:

- Explain the foundational concepts of CI/CD and how they apply in GitLab.
  - Configure basic GitLab pipelines and understand job stages.
  - Use and manage GitLab Runners to execute CI/CD jobs.
  - Implement pipeline variables, artifacts, and basic deployment flows.
  - Apply best practices for CI/CD pipelines in a collaborative team environment.
- 

### Prerequisites

Participants should have:

- Basic Git knowledge (commits, branches, merging).
  - Familiarity with GitLab UI and basic project management.
  - Comfortable with using a terminal or command line.
  - Some experience with DevOps concepts is helpful but not required.
- 

# Training outline

## 1. 1. CI/CD Fundamentals

- What is Continuous Integration?
- What is Continuous Deployment/Delivery?
- Benefits of CI/CD in software workflows.

## 2. 2. Introduction to GitLab CI/CD

- Overview of GitLab's integrated CI/CD.
- How GitLab organizes pipelines and jobs.

## 3. 3. GitLab Runners

- What are GitLab Runners and how they work.
- Shared vs. specific runners.
- Registering and managing a runner.

## 4. 4. Writing Your First Pipeline

- Structure of .gitlab-ci.yml.
- Defining stages and jobs.
- Basic directives and syntax.

## 5. 5. Pipeline Execution and Job Types

- How jobs are executed in stages.
- Understanding successful vs failed jobs.
- Exploring pipeline visualization.

## 6. 6. Variables and Artifacts

- Defining CI/CD variables and secrets.
- Passing artifacts between jobs.
- Using caching to speed up pipelines.

## 7. 7. Basic Deployment Strategies

- Using GitLab for automated deployments.
- Environment targets (staging, production).
- Manual vs automatic job triggers.

## 8. 8. Pipeline Best Practices

- Naming conventions and structure.
- Minimizing pipeline failures.
- Tips for maintainable CI/CD configuration.

## 9. 9. Collaborative CI/CD with GitLab

- Integrating merge requests with pipelines.
- Pipeline status checks before merging.
- Review apps and preview deployments.

## 10. 10. Preparing for Certification (Optional)

- Typical topics tested in *GitLab Certified CI/CD Associate* exam.
- Practice exercises and hands-on labs suggestions.