

DEVOPS

ADVANCED

4 DAYS

Continuous Integration, Delivery, and Deployment with GitLab

Build a strong foundation in CI/CD using GitLab by understanding pipeline principles, configuration, and automation. Learn how to implement automated testing and deployments while leveraging Git version control and Docker containerization for efficient software delivery.

Why GitLab CI/CD?

Streamlined Development Process

GitLab helps to streamline the development process by automating many of the manual tasks. This results in faster, more reliable releases.

Reduced Costs

By automating, GitLab can help to reduce costs associated with development and testing. This can lead to significant cost savings over time.

Improved Collaboration

GitLab provides a centralized platform for collaboration, making it easier for developers, testers, and other stakeholders to work together on a project.

What you'll be able to do

- ✓ Gain a comprehensive understanding of the principles and benefits of CI/CD
- ✓ Learn how to setup the GitLab CI/CD pipeline and its configuration
- ✓ Understand the implementation of automated testing and deployment processes using GitLab CI/CD pipelines
- ✓ Gain command of Version control with Git
- ✓ Learn the basics of Docker containerization

Who this is for

- Software developers
- DevOps engineers
- IT professionals
- Team Leaders

Curriculum

01 CI/CD Foundations

- Apply ci/cd foundations with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

02 GitLab CI/CD Overview

- Apply gitlab ci/cd overview with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

03 Repository and Runner Setup

- Apply repository and runner setup with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

04 .gitlab-ci.yml Pipelines

- Apply .gitlab-ci.yml pipelines with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

05 Automated Testing

- Apply automated testing with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

06 Continuous Deployment

- Apply continuous deployment with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

07 Integrations

- Apply integrations with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

08 Monitoring Pipelines

- Apply monitoring pipelines with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

09 Multi-stage and Artifacts

- Apply multi-stage and artifacts with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

10 Security and Compliance

- Apply security and compliance with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards

11 Best Practices

- Apply best practices with GitLab in practical DevOps workflows
- Complete hands-on labs with realistic scenarios
- Validate outcomes, logs, and failure modes
- Capture reusable patterns for team standards