ONE PAGE SUMMARY

IT & Software › CI/CD › GitLab CI

Introduction to GitLab CI

Streamline your development with GitLab CI for continuous integration and delivery.

DEFINITION

> GitLab CI is a tool that automates the process of testing and deploying code. It helps developers integrate changes more frequently and deliver updates faster.

KEY CONCEPTS

Pipelines

Pipelines are sequences of automated processes that run tests and deployments in GitLab CI.

Jobs

Jobs are individual tasks within a pipeline, such as compiling code or running tests.

Runners

Runners are agents that execute the jobs defined in the pipeline on different environments.

YAML Configuration

GitLab CI uses a YAML file (.gitlab-ci.yml) to define the structure and behavior of pipelines.

Triggers

Triggers are events that start a pipeline, such as code commits or merge requests.

EXAMPLES

→ A pipeline runs tests every time code is pushed to the repository.

→ Deploying an application automatically after successful tests.

→ Using a runner to build a Docker image from the latest code.

MEMORY TIPS

★ Remember 'Pipelines' as a flow of tasks, like water flowing through a pipe.

★ Think of 'Jobs' as individual tasks in a to-do list.

★ Associate 'Runners' with athletes completing tasks quickly and efficiently.

COMMON MISTAKES

✗ Neglecting to define the .gitlab-ci.yml file correctly.

✗ Forgetting to configure runners for job execution.

✗ Overcomplicating pipelines with unnecessary jobs.

QUICK RECAP

GitLab CI automates testing and deployment through pipelines and jobs. Understanding the YAML configuration and how runners work is crucial for effective use. Avoid common mistakes like misconfiguring the .gitlab-ci.yml file.