ONE PAGE SUMMARY

IT & Software › CI/CD › Jenkins

Introduction to Jenkins

Jenkins: Your Gateway to Continuous Integration and Delivery

DEFINITION

> Jenkins is an open-source automation server that helps automate parts of software development related to building, testing, and deploying applications. It enables continuous integration and continuous delivery (CI/CD) to improve software quality and speed up release cycles.

KEY CONCEPTS

Continuous Integration

A development practice where code changes are automatically tested and merged into a shared repository.

Continuous Delivery

An extension of continuous integration that ensures code is always in a deployable state.

Plugins

Jenkins supports various plugins that extend its functionality, allowing integration with different tools and services.

Pipeline

A set of automated processes defined in code that outline the steps for building, testing, and deploying applications.

Master-Slave Architecture

Jenkins can distribute workloads across multiple machines, enhancing performance and scalability.

EXAMPLES

→ Automating the build process for a Java application using Jenkins.

→ Setting up a pipeline to run tests every time code is pushed to the repository.

→ Deploying a web application to a cloud service automatically after successful tests.

MEMORY TIPS

★ Remember 'Jenkins' as 'Just Execute New Knowledge In New Software'.

★ Think of CI/CD as 'Continuous Improvement/Continuous Deployment' to remember its purpose.

★ Visualize a pipeline as a water flow, where each step is a filter for quality.

COMMON MISTAKES

✗ Neglecting to configure proper permissions for Jenkins jobs.

✗ Overcomplicating pipelines with unnecessary steps.

✗ Ignoring plugin updates, which can lead to security vulnerabilities.

QUICK RECAP

Jenkins is a powerful tool for automating software development processes through CI/CD practices. Understanding its key concepts like pipelines and plugins is essential for effective use. Remember to keep your Jenkins environment updated and well-configured to avoid common pitfalls.