

Jenkins Pipeline Scripting (2 Days)

This two-day course takes Jenkins users beyond basic build jobs, providing a comprehensive overview of pipelines. Pipelines support sophisticated continuous integration and continuous delivery processes, using a rich domain-specific language augmented by Groovy programming to automate tasks, even across multiple machines. We cover pipeline automation from the basics through advanced use cases, pointing out common patterns and common pitfalls. A knowledge of basic Jenkins is assumed, but no prior experience with pipelines is required. After this course, you will have the tools to start automating your software build, test, and deployment activities following Jenkins best practices.

Course Outline

Chapter 1 Jenkins essentials refresh

- Review of the fundamentals of Jenkins; baseline knowledge to study pipelines.
- Continuous integration, continuous delivery
- Jenkins as orchestration for build tasks
- Jenkins job types
- Scheduling and triggering jobs
- Security

Chapter 2 Pipeline concepts

- Role for pipelines
- Declarative and scripted pipelines
- Declarative pipeline end to end example
- Scripted pipeline end to end example

Chapter 3 Pipeline domain specific language

- Agents and nodes
- Stages and steps
- Workspaces
- Shell and tools
- Credentials and secret management
- User input
- Flow control
- Stashing and archiving
- Notification

Chapter 4 Distributed builds

- Designing for scale
- Controlling where jobs run
- Configuration management

- Considerations for distributed builds
- State management
- Concurrency concerns
- Latency concerns

Chapter 5 Selected plugins

- Git, GitHub, GitLab, Bitbucket
- Gerrit
- Artifactory
- Jira
- Sonar

Chapter 6 Groovy for pipelines

- Groovy and Pipeline Groovy
- How pipeline Groovy works
- @NonCPS annotation
- Good practices for pipeline code

Chapter 7 Libraries

- Role of libraries
- Defining libraries
- Library structure
- Using libraries

Chapter 8 Extending pipelines with Groovy

- Groovy syntax
- Data types
- Operators
- Flow control
- Try/catch and exceptions
- Functions
- Closures
- Classes

Chapter 9 Best practices

- A collection of best practices for writing and using pipelines. Some common mistakes and how to avoid them.