Introduction to Apache Maven (2 Days)

This Introduction to Apache Maven course starts by helping participants understand the tremendous benefits and return on investment from Apache Maven. After completing this course participants will understand how to download, install, set-up and confidently use core Maven features. The most popular version of the course is taught with Eclipse but it can be delivered with any Integrated Development Environment (IDE).

Objectives

In this training, attendees will learn how to:

- Download and install Maven
- Build a project
- Work with Maven's directory structure, plugins, repositories, and more
- Understand the Project Object Model (POM)
- Build a complete web application using Maven
- Build and activate profiles
- Use Maven from Eclipse via the m2eclipse plugin

Audience

This course is targeted to people who will use Maven to build projects and manage product lifecycles using Maven. They may work on projects that have already been created (i.e. existing project checked out from an SCM system), create new projects based on a Maven archetypes, or produce POM files from scratch. Occasionally they may need to create plugins and POM files that will be referenced by other developers, and to design and manage the organization's repositories.

Prerequisites

All attendees must have core Java® and Java web programming experience.

Course Outline

Chapter 1. Introduction to Apache Maven

- Build Tools for Java
- Build Tools for Java (cont'd)
- History of Build Tools
- Traditional Scripting
- 'make'
- Problems with Make
- Manual Build with JavaC
- ANT
- Pros and Cons of Ant
- Apache Maven
- Goals of Maven
- What is Apache Maven?
- What is Apache Maven (cont'd)
- Why Use Apache Maven?
- The Maven EcoSystem
- Consistent Easy-to-Understand Project Layout
- Convention Over Configuration
- Maven is Different
- Maven Projects have a Standardized Build
- Effect of Convention Over Configuration
- Importance of Plugins
- A Key Point on Maven!

Chapter 2. Installing and Running Apache Maven

- Downloading Maven
- Installing Maven
- Run From Command Line
- Running Inside an IDE
- Settings.xml
- Local Repository

Chapter 3. Getting Started With Maven

- Terminology and Basic Concepts
- Artifacts
- Lifecycle
- Default Lifecycle
- Plugins
- Running Maven - the Story So Far
- Running Maven from an IDE
- Common Goals
- pom.xml
- Example
- Example (cont'd)
- Artifact Coordinates
- Standard Layout for Sources

Chapter 4. A Web Application in Maven

- A More Complex Project
- Putting it Together With Maven
- Packaging the Target Artifact
- The Source Tree
- Dependencies
- Transitive Dependencies
- Dependency Scope
- Working With Servers
- Declaring and Configuring Plugins
- Running the Plugin
- Binding a Plugin Goal to the Lifecycle
- Archetypes

Chapter 5. Commonly Used Plugins

- Maven Plugins
- Declaring and Configuring Plugins
- Running the Plugin
- Binding a Plugin Goal to the Lifecycle
- Maven Surefire Test Plugin
- Failsafe Plugin
- Site Plugin
- JavaDoc Plugin
- PMD Plugin
- Code Coverage – Cobertura

Chapter 6. Multi-Module Builds

- Introduction
- The Reactor
- Reactor Sorting
- Multi-Module Build by Example

Chapter 7. POM Projects

- Project Object Model (POM)
- The overall POM structure
- Storing POM

Chapter 8. Writing Plugins (Maven)

- What is Maven Plugin
- Example of Using a Plugin
- Create a Custom Plugin
- Create a Custom Plugin (cont.)
- Plugin Management

Chapter 9. Creating Archetypes

- Introduction to Maven Archetypes
- Introduction to Maven Archetypes (cont.)
- Using Interactive Mode to generate Goal
- Common Maven Archetypes

Chapter 10. Repository Management

- Maven's Approach to Artifacts
- Publishing Artifacts
- Summary of Maven's Artifact Handling
- Repository
- Repository Manager
- Proxy Remote Repositories
- Types of Artifacts
- Release Artifacts
- Snapshot Artifacts
- Reasons to Use a Repository Manager
- Repository Coordinates
- Addressing Resources in a Repository

Chapter 11. Release Management

- What is release Management?
- Release Management with Nexus
- Release Management with Maven