Introduction to Groovy Programming - TT4630

Gain the latest skills, techniques and best practices required to create robust, well designed and efficient modern Groovy applications

Duration: 3 Days
Skill Level: Introductory
Available Format: Instructor-Led Online ; On Public Schedule

Introduction to Groovy Programming is a dynamic 3-day course designed for OO developers eager to expand their skill set into the versatile world of Groovy.

What You'll Learn

Overview

Introduction to Groovy Programming is a dynamic 3-day course designed for OO developers eager to expand their skill set into the versatile world of Groovy. This high-level language, known for its seamless integration with the Java ecosystem, offers simplified syntax and powerful features, making it valuable for a wide range of applications. From enhancing web development processes to streamlining data analysis, mastering Groovy equips you with the tools to improve productivity and foster innovation in your organization.

The course is structured to provide a comprehensive understanding of Groovy, starting with its role in the Java ecosystem and moving through its practical applications in real-world scenarios. You'll explore Groovy's rich feature set, including advanced data types, control structures, and string manipulation, along with the essentials of object-oriented programming, metaprogramming, and RESTful services integration. These concepts are reinforced through hands-on labs and group activities, where you'll apply what you've learned in realistic situations. Under the guidance of an industry expert, you'll gain practical advice and insights, enhancing your problem-solving skills and building confidence in applying your new knowledge.

You'll exit this course with a robust understanding of Groovy programming, equipped with practical skills and techniques that can be immediately applied to enhance your development projects, along with confidence in navigating complex coding challenges and implementing efficient, innovative solutions in your professional role.

Objectives

This course combines engaging instructor-led presentations and useful demonstrations with valuable real-world hands-on labs and engaging group activities. Throughout the course you'll:

- **Mastery of Groovy Syntax and Language Features:** Gain a thorough understanding of Groovy's syntax, data types, control structures, and operators, enabling you to write more efficient and readable code.
- **Advanced String Manipulation and Regular Expression Skills:** Develop expertise in handling strings and using regular expressions, crucial for data parsing, pattern matching, and text processing tasks in various programming scenarios.
- **Proficiency in Collections and Closures:** Learn to effectively use Groovy's collections (lists, maps, ranges) and closures, enhancing your ability to manage data structures and implement functional programming techniques.
- **Object-Oriented Programming Techniques in Groovy:** Acquire skills in defining classes, methods, and applying concepts of inheritance and polymorphism, essential for developing robust and scalable applications.
- **Skills in Metaprogramming and Integrating Groovy with Java Frameworks:** Understand and apply Groovy's metaprogramming capabilities and AST transformations, along with integrating Groovy with popular Java frameworks like Spring Boot, to build sophisticated applications.
- **Practical Debugging and Problem-Solving Abilities:** Develop strong debugging skills, learning to troubleshoot and resolve common programming issues in Groovy, which enhances your overall problem-solving abilities and prepares you to tackle complex challenges in your development work.

If your team requires different topics, additional skills or a custom approach, our team will collaborate with you to adjust the course to focus on your specific learning objectives and goals.

Audience

The ideal audience for this introductory-level course includes experienced software developers, especially those familiar with object-oriented programming concepts, especially for Java developers looking to enhance their skillset with Groovy's dynamic features. System architects and senior programmers (who have Java experience) seeking to streamline their development process or integrate Groovy into existing Java projects would find this course useful.

Pre-Requisites

To ensure a smooth learning experience and maximize the benefits of attending this course, you should have

- **Basic Object-Oriented Programming Knowledge:** Familiarity with object-oriented programming concepts such as classes, objects, inheritance, and polymorphism is essential, as Groovy builds upon these fundamentals.
- **Experience with Java Development:** A working knowledge of Java is highly beneficial, given Groovy's integration with the Java platform and its similar syntax and libraries.
- **Understanding of Basic Scripting and Automation:** Prior experience with scripting languages or automation tasks can be helpful, as Groovy is often used for scripting within the Java ecosystem.

Agenda

Please note that this list of topics is based on our standard course offering, evolved from typical industry uses and trends. We'll work with you to tune this course and level of coverage to target the skills you need most. Topics, agenda and labs are subject to change, and may adjust during live delivery based on audience skill level, interests and participation.

Day 1: Getting Started with Groovy

Introduction to Groovy
- Overview of Groovy and its place in the Java ecosystem
- Key features and benefits of using Groovy
- Real-world applications and case studies

Setting Up the Environment

- Installing Java JDK: Step-by-step guide
- Groovy installation and configuration
- Introduction to SDKMAN for Groovy management

Groovy Development Tools

- Navigating the Groovy Console: Basic operations and shortcuts
- Utilizing groovysh for interactive scripting
- Compiling scripts with groovyc: Tips and best practices

Getting Hands-On

- Writing your first Groovy script: Syntax essentials
- Basic data types and variable declarations
- Simple input/output operations in Groovy

Day 2: Groovy Language Features

Data Types, Control Structures, and Operators

- Deep dive into Groovy data types
- Understanding Groovy's operator overloading
- Decision-making and control structures: if-else, switch-case

String Manipulation and Regular Expressions

- Effective string handling techniques
- Utilizing regular expressions for pattern matching and text processing
- Practical exercises on strings and regex

Collections and Closures

- Exploring lists, maps, and range objects

- Advanced collection methods and Groovy enhancements
- Introduction to closures: Definition, syntax, and practical uses

Control Structures in Depth
- Loops and iteration: for, while, and advanced looping techniques
- Exception handling in Groovy: try-catch-finally blocks
- Case study: Practical examples of control structures in action

Day 3: Advanced Groovy Concepts and Applications

Object-Oriented Programming in Groovy
- Defining classes, fields, and methods
- Understanding inheritance and polymorphism in Groovy
- Package organization and modular programming

Metaprogramming and AST Transformations
- Concepts of runtime metaprogramming
- AST transformations: Enhancing Groovy at compile-time
- Real-life examples of metaprogramming applications

Groovy Builders and I/O Operations
- Working with MarkupBuilder and JSON Builder for easy data representation
- File input/output best practices
- Introduction to database programming and JDBC integration

REST APIs and Spring Boot Integration
- Developing RESTful services with Groovy
- Best practices for API development and documentation
- Integrating Groovy with Spring Boot: Configuration and setup

Debugging and Final Project
- Debugging techniques in Groovy
- IDE and Tool-Based Debugging
- Advanced Debugging Practices
- Troubleshooting Scenarios

Setup Made Simple! Learning Experience Platform (LXP)

All applicable course software, digital courseware files or course notes, labs, data sets and solutions, live coaching support channels, CodeCoach.AI anytime tutor access, and rich extended learning and post training resources are provided for you in our â€œeasy access, single source, no install requiredâ€ online **Learning Experience Platform (LXP),** remote lab and content environment. Access periods vary by course. Weâ€™ll collaborate with you to ensure your team is set up and ready to go well in advance of the class. Please inquire about set up details and options for your specific course of interest.

For More Information

Please contact us or call 844-475-4559 toll free for more information about our training services (instructor-led, self-paced or blended), coaching and mentoring services, public course enrollment or questions, partner programs, courseware licensing options and more.