

Groovy and Grails

Course Summary

Description

Many have heard the buzz around Grails (a full-stack web-app platform that “attempts to solve as many pieces of the web development puzzle”) and the Groovy language (since 2004, one of the most powerful dynamic languages for the JVM). However, how many can leverage the full power of the platform?

Maybe you've heard about frameworks such as Ruby on Rails, Django or TurboGears and would like to achieve similar benefits in your development shop? Don't let DHH and the Rails community have all the fun...

This workshop is intended to be a solid and pragmatic introduction to using Grails & Groovy at work, or for those advocating or promoting such rapid development frameworks to management. Read on to learn more!

Objectives

By the end of this course, students will be able to:

- Write scripts and full-fledged programs in Groovy
- Integrate with standard Java libraries
- Use Grails as a web-application platform and/or to deliver RESTful web services
- Use Grails scaffolding to build rapid domain-driven prototypes
- Go beyond Grails scaffolding to deliver modular views, using SiteMesh
- Take advantage of Grails AJAX features and tag libraries. Integrate w/ micro-frameworks such as Backbone.js, AngularJS
- Test your code to ensure you and fellow developers adhere to readability and quality standards

Topics

- Introducing Grails: Bringing back the fun to development on the JVM
- Building Your App: Domain-driven development and view scaffolding
- Advanced Views: Templates, dashboards, and the new world of client-side Javascript
- Getting Your App Ready for Staging and Production
- Wrap-up and Review In-Class App Project

Audience

The condensed, one week course is designed for those already familiar with basics of Java SE and web-app development. Perhaps you've worked with Spring or Java Server Faces? Experience on Java Enterprise edition or with the Spring Framework would be especially helpful.

Prerequisites

To take this course, you should know how to code (and test) in Java. You should be able to answer most of the following questions:

- What is a difference between a class and an object?
- What is the difference between static and non-static field?
- What is the difference between extends and implements keywords?
- What is an anonymous inner class? What is the purpose of @Override?

Duration

Five days

Groovy and Grails

Course Outline

- I. Introducing Grails: Bringing back the fun to development on the JVM**
 - A. Introducing Grails
 - B. Meta framework - built on the shoulders of giants: Spring, Hibernate, SiteMesh, Tomcat
 - C. Project Structure
 - D. Convention over configuration -- building w/ opinionated software
 - E. Separation of concerns - Web MVC, etc.

- II. Building Your App: Domain-driven development and view scaffolding**
 - A. Domain objects and GORM (intro.)
 - B. GORM "The Right Relationship" - Objects and Associations
 - C. Scaffolding the UI
 - D. Introducing Groovy: A Robust, Dynamic Language on the JVM w/ many benefits of functional languages

- III. Advanced Views: Templates, dashboards, and the new world of client-side Javascript**
 - A. Controllers and User-friendly Routes
 - B. Continuing the GSP evolution - using and custom Taglibs (MUCH easier than in Spring!)
 - C. Composing complex views and flows
 - D. Alternative view technologies (Grails and MicroJS - Exposing REST services for select Domain(s))

- IV. Getting Your App Ready for Staging and Production**
 - A. Testing at all phases (unit / integration / functional / api)
 - B. Database Migrations
 - C. Adding capabilities thru Plug-ins
 - D. External Configuration / Settings
 - E. Internationalization - loading Messages dynamically in DB

- V. Wrap-up and Review In-Class App Project**
 - A. Introducing Grails in the Enterprise Reverse Engineering
 - B. Modernizing Legacy Apps Grails and MicroJS
 - C. Exposing REST services for select domain(s)
 - D. Multiple DataSource Definitions
 - E. Asynchronous Processing