RESTful Services, Part 4 of 5: Install, JAX-RS, and Jersey

page 1

Meet the expert: Jordan Hudgens has certifications for Ruby on Rails, Thinkful; Ruby on Rails, Bloc.io; Front End Development, Thinkful; and AngularJS, Thinkful. He is currently vice president of engineering for TRACKR in Midland, Texas and is working on his PhD in Computer Science from Texas Tech. In addition to Ruby, Jordan works with PHP, JavaScript, MySQL, Postgres, CSS3, C, C++, C#, Objective-C, and Python. He also works with the frameworks Rails (Ruby), Zend (PHP), and Django (Python), plus the libraries AngularJS, jQuery, and Backbone.js.

Prerequisites: This course assumes that the users have experience coding with Java and RESTful Services. You should have already viewed the “RESTful Services: Overview and JAX-RS,” “RESTful Services: Injection and Sub Resources” and “RESTful Services: Response, Form and Exceptions” courses before viewing this course if you are not familiar with RESTful Services.

Runtime: 01:43:34

Course description: In this course you’ll see a brief history of RESTful Web Services and how they are integral to modern web applications. You’ll see how to setup your computer to begin developing JAX-RS and Jersey applications, specifically by leveraging Maven scaffold applications from the command line.

Course outline:

Dependency Installations
- Introduction
- Environment Configuration
- Demo: Installation
- Jersey 2.15
- Demo: Jersey Setup
- Maven
- Demo: Install Maven
- Demo: Maven Path
- Summary

Deployment Installations
- Introduction
- Git
- Demo: Get Git
- Heroku Toolbelt
- Demo: Heroku
- Summary

Background
- Introduction
- A Primer on JAX-RS & Jersey
- Background
- Properties of RESTful Systems
- Summary

Jax-RS
- Introduction
- Resources
- The Goal of RESTful Services
- RESTful System Actions
- JAX-RS Annotations
- Summary

Jersey
- Introduction
- Where Does Jersey Come In?
- Jersey Features
- Summary

Creating
- Introduction
- Maven Archetypes
- ArtifactId
- Project Version
- groupID
- ArtifactID
- Version
- Package
- Create New Maven Project
- Demo: Create Maven Project
- Summary

Running
- Introduction
- Navigating the Directory
- src/main/java
- Demo: Directory Structure
- Demo: Test File
- Demo; Main Files
- Demo: MyResource
- Navigating to Your App
- Demo: Starting the Application
- Demo: Running the Application
- Summary

Deploying To Heroku
- Introduction
- Create a Project to Deploy
- Demo: Deploy to Heroku
- Demo: Heroku Files
- Demo: Push to Heroku
- Summary