

Instructor Notes:



The slide features a large blue rectangular area at the top. Below this area, the text "Basic Spring 4.0" is displayed in a large, white, sans-serif font, followed by "Lesson 00:" in a smaller, white, sans-serif font. The bottom section of the slide is white and contains the Capgemini logo on the left, the slogan "People matter, results count." in the center, and the Capgemini logo on the right. At the very bottom, there is a small copyright notice: "Copyright © Capgemini 2016. All Rights Reserved" followed by a small number "1".

Basic Spring 4.0
Lesson 00:

People matter, results count.
Capgemini
CONSULTING. TECHNOLOGY. OUTSOURCING

Capgemini
CONSULTING. TECHNOLOGY. OUTSOURCING

Copyright © Capgemini 2016. All Rights Reserved 1

©2016 Capgemini. All rights reserved.
The information contained in this document is proprietary and confidential. For Capgemini only.

Instructor Notes:

Course Goals and Non Goals

- Course Goals

- Understand the benefits of using Spring
- Understand the principles of IoC and AOP
- Be able to use AOP to handle cross-cutting concerns
- Connect business objects to persistent stores using Spring's DAO modules
- Use the Spring MVC web framework to develop flexible web applications
- Introduction to Spring Testing

- Course Non Goals

- Design patterns, Spring Integration with different technologies

Instructor Notes:

Pre-requisites

- Core Java , Java 8 features and JDBC
- XML, DBMS/SQL
- Servlets, JSP
- Concepts of MVC, Design patterns

Instructor Notes:

Intended Audience

- All Java application developers especially Enterprise Java Programmers
- Software designers



Instructor Notes:

Day Wise Schedule

- Day 1
 - Lesson 1: Introduction to Spring Framework, IoC
- Day 2
 - Lesson 2 : Spring MVC framework
- Day 3
 - Lesson 3: Spring JPA Integration

Instructor Notes:

Table of Contents

- Lesson 1: Introduction to Spring Framework, IoC
 - 1.1 What is Spring Framework, Benefits of Spring
 - 1.2 The Spring architecture
 - 1.3 IOC – Inversion of control, wiring beans
 - 1.4 Bean containers, lifecycle of beans in containers
 - 1.5 Customizing beans with PostProcessors
 - 1.6 Annotation-based configuration
- Lesson 2: Spring MVC framework
 - 2.1 Introduction: DispatcherServlet, Handler mappings, Resolving views
 - 2.2 Annotation-based controller configuration
 - 2.3 Introduction to REST web Services
 - 2.4 REST Controllers on the top of MVC

Instructor Notes:

Table of Contents

- Lesson 3: Spring JPA Integration
 - 3.1 Spring support for JPA
 - 3.2 Implementing Spring JPA integration
 - 3.3 Spring Data JPA

Instructor Notes:

References

- Spring in Action, Fourth Edition, Manning publications by Craig Walls
- Spring-framework-reference.pdf from SpringSource (this is available in the downloaded Spring software)



Instructor Notes:

Software required

- JDK version 1.8 + with help, Netscape or IE
- MS-Access/Connectivity to Oracle database
- WildFly
- Eclipse Luna
- Spring 4.0 API with docs

Instructor Notes:

Other Parallel Technology Areas

- EJB 3.0
- PicoContainer
- NanoContainer
- Keel Framework
- Google Guice



Copyright © Capgemini 2015. All Rights Reserved 11

- PicoContainer: is an exceptionally small DI (Dependency Injection) container that allows to use DI for your application without introducing any dependencies other than PicoContainer itself
- NanoContainer: is an extension to PicoContainer for managing trees of individual PicoContainer containers.
- Keel Framework: is more of a metaframework, in that most of its abilities come from other frameworks that are all brought together under one roof.
- Google Guice: focuses purely on DI.