

Copyright © 2016 by The Learning House.

All rights reserved. No part of these materials may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of The Learning House. For permission requests, write to The Learning House, addressed “Attention: Permissions Coordinator,” at the address below.

The Learning House  
427 S. 4<sup>th</sup> Street #300  
Louisville KY 40202



---

# Web Apps and Spring Core MVC Unit

## Lesson 3 – Java Server Pages

# Objectives

- Understand where JSPs fit into a Java web app
- Understand how to integrate JSPs with Servlets
- Understand JSP Expression Language (EL) basics
- Understand JSTL tag lib basics

# What are JSPs

- A JSP is a mix of HTML and special Java-related tags
- You can write regular Java in a JSP but you should NEVER do this
- JSPs become servlets via container translation
- JSPs are the View in the Model-View-Controller model

# Lifecycle of a JSP

1. You write and save the JSP and deploy your application
2. The container will translate the JSP to Java
3. The container will compile the .java file to a .class file (catching syntax errors here). The .class file is a compiled servlet
4. The container instantiates the new servlet class
5. The will direct requests the the new servlet as with any servlet

# EL

- Solves 2 big problems:
  - Web designers don't (and shouldn't have to) know Java
  - Java code in JSPs is very hard to maintain
- Expressions always take the form:
  - `${something}`
- We can access properties of POJOs (beans) with the `.` operator
  - `person.name` is like `person.getName()` in Java code

# JSTL

- A tag library that allows you to put conditions and loops in your JSP
- Useful for displaying lists of information to the screen
- Remember that the logic you put in JSPs should only be View related - all business logic should occur on the server

# Examples

- RSVP with Servlets and JSPs
- Simple Employee Contact Info page