

# Spring MVC 4 Controllers

Bryan Hansen  
twitter: bh5k

<http://www.linkedin.com/in/hansenbryan>



**pluralsight** hardcore dev and IT training

# What Is a Controller?



# Responsibilities

- Interpret user input and transform that input to a model
- Provide access to business logic
- Determines view based off of logic
- Interprets Exceptions from the business logic / service tier



# Controller Annotations

- **@Controller**
  - Works like it did in Spring MVC 3
- **@RestController**
  - New to Spring MVC 4
- **@Configuration**
  - Signifies a configuration class
- **@EnableWebMvc**
  - Enables our Java configuration
- **@ComponentScan**
  - Override the default scan location for controllers



# Enable Web MVC

- Convenience annotation for `WebMvcConfigurationSupport`
- Only used for Java configuration of Spring MVC web apps
- Customizable by extending `WebMvcConfigurerAdapter`



# Component Scan

- Component scan in XML:

```
<context:component-scan base-package="com.pluralsight"/>
```

- Component scan in Java Config:

```
@ComponentScan(basePackages = "com.pluralsight")
```

# @Controller

```
@Controller  
public class HelloController {  
  
    @RequestMapping(value = "/greeting")  
    public String sayHello (Model model) {
```

# **web.xml**

- Using a web.xml file doesn't abandon Java configuration
- Still need a Dispatcher Servlet
- contextConfigurationLocation points to class rather than XML file



# Without web.xml

- Still need a mapping somewhere
- WebApplicationInitializer
  - Servlet 3.0 hooks
- Builds ApplicationContext

<zm1>

# Summary

- Controllers
- Annotations
- EnableWebMvc
- Component Scanner
- web.xml
- Without web.xml