

If you follow my 3.2 course using Spring 4, as far as I know there are no changes to the non-web parts of the course, except you'll be using Spring 4 dependencies instead of 3.2, and will need the corresponding headers for your XML files.

One small difference is that at once point I show that trying to cast a proxied bean gives an error. But the default proxy type for Spring 4.2 is now cglib, and this type of proxy doesn't give any error; instead of wrapping your classes in another class, cglib works at a lower level.

You'll need a different log out url, in a "post" form, when you get to the bit about creating Spring web login forms.

```
<c:url var="logoutUrl" value="/logout"/>
<form action="{logoutUrl}" method="post">
<input type="submit" value="Log out"/> <input type="hidden"
name="{_csrf.parameterName}" value="{_csrf.token}"/>
</form>
```

The login url has also change, to "login", so your login form should look like this:

```
<c:url value="/login" var="loginUrl"/>
<form action="{loginUrl}" method="post">
....
```

When you incorporate Javascript with Spring, you'll also need to either disable CSRF protection, or else add a couple of lines to the header of templates/default.jsp in the <head> section:

```
<!-- default header name is X-CSRF-TOKEN -->
<meta name="_csrf_header" content="{_csrf.headerName}"/>
```

In home.jsp there will be an onLoad function eventually in the course, which has to be changed to look like this:

```
function onLoad() {

    var token = $("input[name='_csrf']").val();
    var header = "X-CSRF-TOKEN";
    $(document).ajaxSend(function(e, xhr, options) {
        xhr.setRequestHeader(header, token);
    });

    updatePage();
    startTimer();
}
```

You also need a newer dependency for the Jackson library for Javascript JSON:

```
<dependency>
  <groupId>com.fasterxml.jackson.core</groupId>
  <artifactId>jackson-databind</artifactId>
  <version>2.5.1</version>
</dependency>
```

When you add roles for users, they must begin with ROLE_; e.g. ROLE_ADMIN

And finally, there's a bug in the code for the final project in this course which sometimes causes a redirection to the JSON service page on login. You can fix that by adding `always-use_default_target` to your `security:form-login` xml:

```
<security:form-login always-use-default-target="true" login-page="/login"
  default-target-url="/" authentication-failure-url="/login?error=true" />
```