All Tests (https://rank.sanfoundry.com/spring-tests/)

Top Rankers (https://rank.sanfoundry.com/top-rankers-spring/)

Practice Test 1 (https://rank.sanfoundry.com/spring-practice-test-1/)

Practice Test 2 (https://rank.sanfoundry.com/spring-practice-test-2/)

Practice Test 3 (https://rank.sanfoundry.com/spring-practice-test-3/)

Practice Test 4 (https://rank.sanfoundry.com/spring-practice-test-4/)

Practice Test 5 (https://rank.sanfoundry.com/spring-practice-test-5/)

Practice Test 6 (https://rank.sanfoundry.com/spring-practice-test-6/)

Practice Test 7 (https://rank.sanfoundry.com/spring-practice-test-7/)

Practice Test 8 (https://rank.sanfoundry.com/spring-practice-test-8/)

Practice Test 9 (https://rank.sanfoundry.com/spring-practice-test-9/)

Practice Test 10 (https://rank.sanfoundry.com/spring-practice-test-10/)

Mock Test 1 (https://rank.sanfoundry.com/spring-mock-test-1/)

Mock Test 2 (https://rank.sanfoundry.com/spring-mock-test-2/)

Mock Test 3 (https://rank.sanfoundry.com/spring-mock-test-3/)

Mock Test 4 (https://rank.sanfoundry.com/spring-mock-test-4/)

Mock Test 5 (https://rank.sanfoundry.com/spring-mock-test-5/)

Mock Test 6 (https://rank.sanfoundry.com/spring-mock-test-6/)

Mock Test 7 (https://rank.sanfoundry.com/spring-mock-test-7/)

Mock Test 8 (https://rank.sanfoundry.com/spring-mock-test-8/)

Mock Test 9 (https://rank.sanfoundry.com/spring-mock-test-9/)

Mock Test 10 (https://rank.sanfoundry.com/spring-mock-test-10/)

« Prev Page (https://www.sanfoundry.com/spring-interview-questions-answers-freshers/)

Next Page (https://www.sanfoundry.com/spring-questions-answers-bean-validation-creating-excel-pdf/) »

Spring Questions and Answers – Views Mapping and Handling

This set of Java Spring Multiple Choice Questions & Answers (MCQs) focuses on “Views Mapping and Handling”.

1. Interface for Dispatcher Servlet to auto detect view resolver beans.
a) localeResolver
b) tiles
c) ViewResolver
d) none of the mentioned

View Answer

Answer: c
Explanation: Spring MVC comes with several ViewResolver implementations for you to resolve views using different strategies.

2. ViewResolver which maps each view name tol an application's directory.

a) InternalViewResolver

b) InternalResourceViewResolver

c) InternalResolver

d) None of the mentioned

View Answer

Answer: b

Explanation: The view resolver InternalResourceViewResolver maps each view name to an application's directory by means of a prefix and a suffix declaration.

3. InternalResourceViewResolver resolves view names into view objects of type.

a) JstlView

b) Jst Library

c) View

d) None of the mentioned

View Answer

Answer: a

Explanation: By default, InternalResourceViewResolver resolves view names into view objects of type JstlView if the JSTL library (i.e., jstl.jar) is present in the classpath.

4. By default, XmlViewResolver loads view beans from /WEB-INF/views.xml which is final.

a) True

b) False

View Answer

Answer: b

Explanation: This location can be overridden through the location property.

5. View beans from a resource bundle is loaded by.

a) ResourceBundleViewResolver

b) ResourceBundleView

c) ResourceViewResolver

d) None of the mentioned

View Answer

Answer: a

Explanation: ResourceBundleViewResolver loads view beans from a resource bundle in the classpath root.

6. Views that can't be resolved by InternalResourceViewResolver.

a) redirect

b) redirect:prefix

c) redirect:suffix

d) all of the mentioned

View Answer

Answer: a

Explanation: Now the resource bundle views.properties should only contain the views that can't be resolved by InternalResourceViewResolver (e.g., the redirect views).

7. Properties used to determine the correct content and type for a certain web requests.
a) The URL extension provided in a request
b) The HTTP Accept header
c) All of the mentioned
d) None of the mentioned

View Answer

Answer: c
Explanation: When a request is received for a web application, it contains a series of properties that allow the processing framework, in this case Spring MVC, to determine the correct content and type to return to the requesting party.

8. Inspection of header is done by which of the following.
a) InternalViewResolver
b) ContentNegotiatingViewResolver
c) ContentNegotiating
d) InternalResolver

View Answer

Answer: b
Explanation: Inspecting HTTP Accept headers in a controller can be a messy process. So Spring MVC supports the inspection of headers through the ContentNegotiatingViewResolver.

9. ContentNegotiatingViewResolver resolver configuration sets up the resolver to have the highest priority among all resolvers.
a) True
b) False

View Answer

Answer: a
Explanation: It's necessary to make the content negotiating resolver work.

10. The ContentNegotiatingViewResolver resolver first determines the media type for a request based on which of the following criteria?
a) It checks a request path extension
b) If a request path has an extension but no match can be found in the default mediaTypes section of the ContentNegotiatingViewResolver bean
c) If no extension is present in a request path, the HTTP Accept header of the request is used
d) All of the mentioned

View Answer

Answer: d
Explanation: It checks a request path extension (e.g., .html, .xml, or .pdf) against the default
media types (e.g., text/html) specified in the mediaTypes section of the
ContentNegotiatingViewResolver bean.
If a request path has an extension but no match can be found in the default
mediaTypes section of the ContentNegotiatingViewResolver bean, an attempt is
made to determine an extension's media type using FileTypeMap belonging to Java

Activation Framework.
If no extension is present in a request path, the HTTP Accept header of the request is used.

11. In a Spring MVC application, you can register one or more exception resolver beans in the web application context to resolve uncaught exceptions.

a) True

b) False

View Answer

Answer: a

Explanation: These beans have to implement the HandlerExceptionResolver interface for DispatcherServlet to auto-detect them.

12. To configure the exception mappings in the web application context.

a) SimpleMappingExceptionResolver

b) ExceptionResolver

c) ExceptionViewResolver

d) None of the mentioned

View Answer

Answer: a

Explanation: Spring MVC comes with the exception resolver SimpleMappingExceptionResolver for you to configure the exception mappings in the web application context.

13. Annotation which allows a controller’s field to be assigned using Spring Expression Language(SpEL)

a) @Value

b) @After

c) @Default

d) @None

View Answer

Answer: a

Explanation: You can use the @Value annotation along with SpEL to query beans present in an application’s context and extract values to help you initialize controller fields.

14. SpEL statements are recognizable.

a) True

b) False

View Answer

Answer: a

Explanation: They use a notation in the form “#{ SpEL statement }”.

15. Tag, used to define a location in which to place errors in case a form.

a) form

b) form:GET

c) form:POST

d) form:errors

View Answer

Answer: d

Explanation: You can find the form:errors tag, used to define a location in which to place errors in case a form does not meet the rules set forth by a controller.

Sanfoundry Global Education & Learning Series – Java Spring.

To practice all areas of Java Spring, here is complete set of 1000+ Multiple Choice Questions and Answers (https://www.sanfoundry.com/1000-spring-questions-answers/).

« Prev Page - Spring Questions and Answers – User Locales and Locale-Sensitive Text Messages (https://www.sanfoundry.com/spring-interview-questions-answers-freshers/)
» Next Page - Spring Questions and Answers – Bean validation and creating excel and pdf (https://www.sanfoundry.com/spring-questions-answers-bean-validation-creating-excel-pdf/)

Deep Dive @ Sanfoundry:

1. **Database Management System Questions and Answers (https://www.sanfoundry.com/1000-database-management-system-questions-answers/)**
2. **Java Questions and Answers (https://www.sanfoundry.com/java-questions-answers-freshers-experienced/)**
3. **RDBMS Questions and Answers (https://www.sanfoundry.com/1000-rdbms-questions-answers/)**
4. **Java Programming Examples on String Handling (https://www.sanfoundry.com/java-programming-examples-string-handling/)**
5. **C Programming Examples on File Handling (https://www.sanfoundry.com/c-programming-examples-file-handling/)**
6. **Java Programming Examples on Event Handling (https://www.sanfoundry.com/java-programming-examples-event-handling/)**
7. **Java Programming Examples on Exception Handling (https://www.sanfoundry.com/java-programming-examples-exception-handling/)**
8. **Java Programming Examples on File Handling (https://www.sanfoundry.com/java-programming-examples-file-handling/)**
9. **Spring Questions and Answers (https://www.sanfoundry.com/1000-spring-questions-answers/)**
10. **Spring Questions and Answers – Bean Post Processors (https://www.sanfoundry.com/spring-questions-answers-bean-post-processors/)**