# Spring Questions and Answers – REST Service with Spring

Posted on [June 12, 2017](https://www.sanfoundry.com/spring-questions-answers-rest-service-spring/) by [Manish](https://www.sanfoundry.com/author/bhojas/)

This set of Java Spring Multiple Choice Questions & Answers (MCQs) focuses on “REST Service with Spring”.

1. To publish a REST service with Spring.  
a) publishing an application’s data as a REST service  
b) accessing data from third-party REST services  
c) none of the mentioned  
d) all of the mentioned  
View Answer

Answer: d  
Explanation: One involves publishing an application’s data as a REST service, the other one involves accessing data from third-party REST services to be used in an application.

2. Publishing an application’s data as a REST service requires.  
a) @RequestMapping  
b) @PathVariable  
c) All of the mentioned  
d) None of the mentioned  
View Answer

Answer: c  
Explanation: Publishing an application’s data as a REST service revolves around the use of the Spring MVC annotations @RequestMapping and @PathVariable.

3. Spring supports a series of mechanisms to generate a REST service’s payload.  
a) True  
b) False  
View Answer

Answer: a  
Explanation: This recipe will explore the simplest mechanism, which involves the use of Spring’s MarshallingView class.

4. Annotation added as an input parameter to the handler method.  
a) @PathVariable  
b) @Path  
c) @PathLocale  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: The @PathVariable annotation is added as an input parameter to the handler method, per Spring’s MVC conventions, in order for it to be used inside the handler method body.

5. Notation for defining REST end points.  
a) { }  
b) \*  
c) All of the mentioned  
d) None of the mentioned  
View Answer

Answer: c  
Explanation: In addition to supporting the { } notation, it’s also possible to use a wildcard \* notation for defining REST end points.

6. General-purpose class that allows a response to be rendered using a marshaller.  
a) MarshallingView  
b) Marshalling  
c) View  
d) All of the mentioned  
View Answer

Answer: a  
Explanation: The membertemplate view is defined as a MarshallingView type, which is a general-purpose class that allows a response to be rendered using a marshaller.

7. Marshalling is the process of transforming an in-memory representation of an object into a data format.a  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Therefore, for this particular case, a marshaller is charged with transforming a Member object into an XML data format.

8. The marshaller used by MarshallingView belongs to one of a series of XML marshallers.  
a) Jaxb2Marshaller  
b) XmlBeansMarshaller  
c) CastorMarshalle  
d) All of the mentioned  
View Answer

Answer: d  
Explanation: The marshaller used by MarshallingView belongs to one of a series of XML marshallers provided by Spring—Jaxb2Marshaller. Other marshallers provided by Spring include CastorMarshaller, JibxMarshaller, XmlBeansMarshaller, and XStreamMarshaller.

9. To configure Jaxb2Marshaller marshaller we require.  
a) ClassesToBeBound  
b) ContextPath  
c) All of the mentioned  
d) None of the mentioned  
View Answer

Answer: c  
Explanation: The Jaxb2Marshaller marshaller requires to be configured with either a property named classesToBeBound or contextPath.

10. In the case of classesToBeBound, the classes assigned to this property, indicate the class (i.e., object) structure that is to be transformed into XML.  
a) True  
b) False  
View Answer

Answer: a  
Explanation: The Jaxb2Marshaller marshaller requires to be configured with either a property named classesToBeBound or contextPath.

11. Annotation which allows the Jaxb2Marshaller marshaller to detect a class’s (i.e., object’s) fields.  
a) @XmlRootElement  
b) @XmlRoot  
c) @NotNull  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: This annotation allows the Jaxb2Marshaller marshaller to detect a class’s (i.e., object’s) fields and transform them into XML data.

12. Accessing a third-party REST service inside a Spring application.  
a) RestTemplate Class  
b) ViewResolver  
c) InternalViewResolver  
d) View  
View Answer

Answer: a  
Explanation: Accessing a third-party REST service inside a Spring application revolves around the use of the Spring RestTemplate class.

13. REST service end point comprises an address.  
a) starts with http:// and ends with ?  
b) starts with http:// and ends with &  
c) no certain URL is specified  
d) depends upon the platform used  
View Answer

Answer: a  
Explanation: The structure of the REST service end point comprises an address, which starts with http:// and ends with ?, as well as a series of parameters that start with ? and are delimited by &, each represented by a key and value divided by =.

14. XML tag which represents information related to a REST service’s request.  
a) Result  
b) Title  
c) None of the mentioned  
d) All of the mentioned  
View Answer

Answer: d  
Explanation: The actual meaning of the payload is highly dependent on a REST service. In this case, the XML tags (e.g., Result, Title ) are definitions set forth by Yahoo, while the character data enclosed in each XML tag represents information related to a REST service’s request.

15. RestTemplate class method which performs an HTTP HEAD operation.  
a) headForHeaders(String, Object…)h  
b) getForObject(String, Class, Object…)  
c) postForLocation(String, Object, Object…)  
d) postForObject(String, Object, Class, Object…)  
View Answer

Answer: a  
Explanation: headforHeaders() is a RestTemplate class method based on HTTP protocol’s request methods.