1) Controller Class renders the objects passed by the controller’s handler method.

a) True

b) False

Ans:a

2) Class used to give a class behavior of a Dispatcher Servlet.

a) AbstractController

b) Controller

c) Abstract Class

d) AbstractAction

Ans:a

3) All of spring’s various Controller inherit from AbstractControllor?

a. True

b. False

Ans:a

4) WebApplicationContext has more scopes like request, session, global Session to the

standard Application context (which only provides Singleton and Prototype scopes)

a. True

b. False

Ans:a

5) How the Spring Dispatcher Servlet supports for returning the last-modification-date, as

specified by the Servlet API?

A. By implementing the LastModified interface

B. By implementing the LastSessionModified interface

C. By implementing the LastDateModified interface

D. By implementing the LastModifiedDate interface

Ans:d

6) Which of the among is not a special bean in WebApplicationContext?

A. Locale resolver

B. Theme resolver

C. multipart file

D. resolver

E. Handler exception resolver

F. Handler mappings

G. None of the above

Ans:d

7) Design pattern implemented by Dispatcher Servlet.

a) jsp

b) tiles

c) front controller

d) none of the mentioned

Ans:c

8) Which of the following is not a view resolver?

A. BeanNameViewResolver

B. VelocityViewResolver

C. JasperReportsViewResolver

D. UrlBasedViewResolver

E. ResourceBundleViewResolver

F. none of above

Ans:d

9) By default, @RequestMapping gets all the POST Requests.

A. True

B. False

Ans:b

10) Handler method annotation.

a)@Before

b) @Controller

c) @After

d) @RequestMapping

Ans:d

11. Method arguments that can be used in handler methods using the @RequestMapping

annotation.

a HttpServletRequest or HttpServleResponse

b @RequestParam

c @ModelAttribute

d All of the mentioned

ans:d

12 .The @RequestMapping annotation can be applied to which levels.

a class

b methods

c both of the above

d none

Ans:c

13. Where we use Repository annotation?

a. test class

b. dao class

c. bean class

d. controller class

Ans:b

14. Which of the following layers the @Controller annotation is used in?

a. Service layer.

b. Presentation layer.

c. None of these

d. Business layer.

Ans:b

15 Which annotation is used to bind a request parameter a variable in method scope?

a. @ModelAttribute

b. @PathVariable

c. @RequestParam

d. none of the above

Ans:c

16. Which methods are invoked in controller before @RequestMapping methods in same

controller?

a.@RequestParam

b.@ModelAttribute

c. all of the above

d @None of the above

Ans:b

17. Which feature of spring gives us annotation?

a. Transaction management.

b. Inversion of control container (dependency injection)

c. autodetection

d. Data access framework.

Ans:c

18. What is the use of ModelAttribute annotation?

a. injects data objects the model before jsp loads

b. read data from an existing model assigning it to handler method parameters

c. both a and b

d. none of above

Ans:c

19. Identify which annotation can be used with @Controller?

a. @RequestMapping

b. @ModelAttribute

c. @PathVariable

Ans:a

20. Which package is used for Controller annotation?

a. import org.springframework.beans.factory

b. import org.springframework.stereotype.Controller

c. import org.stereotype.Controller

d. import springframework.stereotype.Controller

Ans:b

21. Beans can be created by which of the following properties?

a) Scope

b) Property

c) Class

d) It’s own constructor

Ans:d

22. What will be the output?

public class CreatePro {

String ProductId;

public CreatePro(String ProductId) {

this.ProductId = ProductId;

}

public static Product creation\_Product(String productId) {

System.out.println("Bean Created");

if ("aaa".equals(productId)) {

return new Battery("AAA", 2.5);

} else if ("cdrw".equals(productId)) {

return new Disc("CD-RW", 1.5);

}

}

}

<beans ...>

<bean id="aaa" class="CreatePro"

factory-method="createProduct">

<constructor-arg value="aaa" />

</bean>

<bean id="cdrw" class="CreatePro"

factory-method="createProduct">

<constructor-arg value="cdrw" />

</bean>

</beans>

a) BeanCreationException

b) Bean Created

c) ClassPathException

d) None of the mentioned

Ans:a

23. A bean can have more than one name using multiple id attributes?

a) True

b) False

Ans:a

24. Which one is the default scope of the beans?

a) Prototype

b) Session

c) Request

d) Singleton

Ans:d

25. Alternate way of initialization method is:-

a) init-method attribute

b) afterPropertiesSet

c) destroy-method attribute

d) none of the mentioned

Ans:a

26. Which special type of advice is used to implement an interface?

a) Introduction

b) Before

c) After

d) AfterSpecial

Ans:a

27. Attribute used to specify implementation class

a) no such attribute

b) defaultImpl

c) defaultInterface

d) defaultImplement

Ans:b

28. How to keep track of the usage of your beans

a) Calling Count

b) Last modified date

c) New Interface to your object

d) All of the mentioned

Ans:d

29. Tag used to declare aspects

a) aop:config

b) aop:auto-proxy

c) aop:autowire

d) None of the mentioned

Ans:a

30. Is this bean configuration metadata correct?

<beans ...>

bean id="com.shop.Product.AAA"

class="org.springframework.beans.factory.config.

FieldRetrievingFactoryBean" /

bean id="com.shop.Product.CDRW"

class="org.springframework.beans.factory.config.

FieldRetrievingFactoryBean" /

</beans>

a) Yes

b) No

Ans:a

31. What is used to dispatch requests for web application?

a)ViewHandler

b)ViewResolver

c)DispatcherServlet

d)None Of the above

Ans:c

32. Class used to give a class behavior of a Dispatcher Servlet.

a) AbstractController

b) Controller

c) Abstract Class

d) AbstractAction

Ans:a

33. Design pattern implemented by Dispatcher Servlet.

a) jsp

b) tiles

c) front controller

d) none of the mentioned

Ans:c

34. Interface for Dispatcher Servlet to auto detect view resolver beans.

a) localeResolver

b) tiles

c) ViewResolver

d) none of the mentioned

Ans:c

35. Views that can’t be resolved by InternalResourceViewResolver.

a) redirect

b) redirect:prefix

c) redirect:suffix

d) all of the mentioned

Ans:a

36. Which class or interface is not used while configuring Dispatcher Servlet

programmatically?

a)WebAppInitializer

b)AbstractDispatcherServletInitializer

c)SpringServletContainerInitializer

d)AbstractAnnotationConfigDispatcherServletInitializer

Ans:c

37. Which of the following acts as the front controller in Spring MVC Framework?

a)Spring Bean Classes

b)DispatcherServlet Class

c)Spring Views

d)SpringController Class

Ans:b

38. Which Initialization parameter is not used in Dispatcher Servlet?

a)contextClass

b)contextConfig

c)contextConfigLocation

d)namespace

Ans:b

39. By default,when the DispatcherServlet class is loaded,it loads the\_\_\_\_\_\_\_ from the

XML file.

a)SpringApplicationContext

b)SpringDispatcher

c) SpringController

d) ApplicationContext

Ans:a

40. Which bean does not support Dispatcher Servlet?

a)HandlerMapping

b)ViewResolver

c)HandlerExceptionResolver

d)HandlerResolver

Ans:d

41. Interface for Dispatcher Servlet to auto detect view resolver beans.

a) localeResolver

b) tiles

c) ViewResolver

d) none of the mentioned

Ans:c

42. ViewResolver which maps each view name tol an application’s directory.

a) InternalViewResolver

b) InternalResourceViewResolver

c) InternalResolver

d) None of the mentioned

Ans:b

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

a) JstlView

b) Jst Library

c) View

d) None of the mentioned

Ans:a

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

final.

a) True

b) False

Ans:b

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

a) ResourceBundleViewResolver

b) ResourceBundleView

c) ResourceViewResolver

d) None of the mentioned

Ans:a

46. Views that can’t be resolved by InternalResourceViewResolver.

a) redirect

b) redirect:prefix

c) redirect:suffix

d) all of the mentioned

Ans:a

47. Inspection of header is done by which of the following.

a) InternalViewResolver

b) ContentNegotiatingViewResolver

c) ContentNegotiating

d) InternalResolver

Ans:b

48. ContentNegotiatingViewResolver resolver configuration sets up the resolver to have the

highest priority among all resolvers.

a) True

b) False

Ans:a

49. 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

Ans:d

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

a) SimpleMappingExceptionResolver

b) ExceptionResolver

c) ExceptionViewResolver

d) None of the mentioned

Ans:a

51. Annotation for Controller Class.

a) @Before

b) @Controller

c) @After

d) @Exception

Ans:b

52. Controller Class renders the objects passed by the controller’s handler method.

a) True

b) False

Ans:a

53. import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.springframework.web.servlet.ModelAndView;

import org.springframework.web.servlet.mvc.AbstractController;

public class BigController extends AbstractController

{

@Override

protected ModelAndView handleRequestInternal(HttpServletRequest

request, HttpServletResponse response) throws Exception

{

System.out.println("You're big!");

return new ModelAndView("big");

}

}

This above code creates a single-action controller with configurations regarding

supported

methods, session and caching can be specified in the bean declaration of the controller.

<bean name="/big" class="net.codejava.spring.BigController"

<property name="supportedMethods" value="POST"/>

</bean>

This configuration supports which method that is supported by this controller handler

method

a) Only GET method

b) Only POST method

c) Both a and b

Ans:b

54. After Controller has finished handling a render request, it returns:-

a) model name

b) view name

c) view object

d) all of the mentioned

Ans:d

55. Method arguments that can be used in handler methods using the @RequestMapping

annotation.

a) HttpServletRequest or HttpServleResponse

b) @RequestParam

c) @ModelAttribute

d) All of the mentioned

Ans:d

56. The @RequestMapping annotation can be applied to the class level only.

a) True

b) False

Ans:b

57. By default, @RequestMapping gets all the POST Requests.

a) True

b) False

Ans:b

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

a) ExceptionResolver

b) ExceptionViewResolver

c) SimpleMappingExceptionResolver

d) None of the mentioned

Ans:c

59. \_\_\_\_\_\_\_\_\_\_\_is a specific type of component, used in MVC applications and mostly

used with RequestMapping annotation.

a) @Controller

b) @Component

c) @Repository

d) @Service

Ans:a

60. As per Spring 3.O doc's has \_\_\_\_\_ controllers are in Spring MVC

a) 2

b) 10

c) 4

d) 20

Ans:d

61. Which handler to use, when this handler mapping does not result in a matching handler.

a. default Handler

b. HandlerExecutionChain

c. DispatcherServlet

d.BeanNameURLHandlerMapping

Ans:a

62. Handler Mapping is \_\_\_\_\_\_\_\_\_\_\_

a.an interface that defines a mapping between components

b.an interface that defines a mapping between requests and handler objects.

c.an interface that defines a mapping between responses.

d.an interface that defines a mapping between servlets.

Ans:b

63. BeanNameUrlHandlerMapping maps request URLs to

a. beans with the different name.

b. ​beans with the same name.

c. controller .

d. component.

Ans:b

64. Defining a bean for BeanNameUrlHandlerMapping is not required

a. true

b. false

Ans:a

65. SimpleUrlHandlerMapping

a. allows for direct and declarative mapping between either bean instances and

URLs or between bean names and URLs.

b. Does not allow for direct and declarative mapping between bean instances and

URLs but allows between bean names and URLs.

c. allows for direct and declarative mapping between bean instances and URLs but

not between bean names and URLs.

d. Does not allow for direct and declarative mapping bean instances and URLs or

between bean names and URLs.

Ans:a

66. To get the control over which mapping is used, the priorities are set using

a. constructor .

b. getOrder(int order) method.

c. setOrder(int order) method

d. None

Ans:c

67. A handler will always be wrapped in a \_\_\_\_\_\_\_\_\_\_ instance

a.Handler Mapping

b.BeanNameURLHandlerMapping

c. RequestMappingURLHandlerMapping

d.​ ​HandlerExecutionChain

Ans:d​

68. HandlerExecutionChain throws

a. java.io.\*

b.java.lang.object

c.Java.util.\*;

d. java.lang.Exception

Ans:d

69. If Spring can’t found handler mapping, the DispatcherServlet will creates

a \_\_\_\_\_\_\_\_\_\_\_\_\_ automatically.

a. RequestMappingURLHandlerMapping

b.HandlerExecutionChain

c.​ BeanNameUrlHandlerMapping

d. DispatcherServlet.

Ans:c

70.Handler Mapping is an interface

a. true

b.false

Ans:a