SPRING MOCK TEST
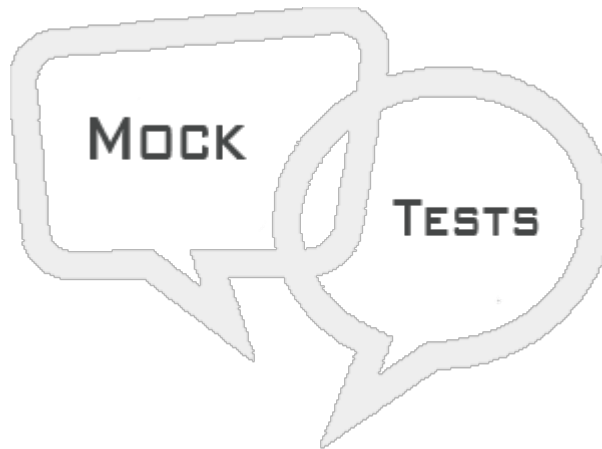
This section presents you various set of Mock Tests related to **Spring Framework**. You can download these sample mock tests at your local machine and solve offline at your convenience. Every mock test is supplied with a mock test key to let you verify the final score and grade yourself.

SPRING MOCK TEST I

Q 1 - What is spring?

A - Spring is an open source development framework for enterprise Java.

B - Spring is a proprietary framework.

C - Spring is a development framework for .Net applications.

D - Spring is a development framework for PHP based applications.

Q 2 - Which of the following is correct assertion about spring?

A - Spring enables developers to develop enterprise-class applications using POJOs.

B - Spring is organized in a modular fashion.

C - Testing an application written with spring is simple because environment-dependent code is moved into this framework.

D - All of above.

Q 3 - What is Dependency Injection?

A - It is a design pattern which implements Inversion of Control for software applications.

B - It is one of the spring module.

C - It is a technique to get dependencies of any project.

D - It is used to promote tight coupling in code.

Q 4 - Which of the following is correct about dependency injection?

A - It helps in decoupling application objects from each other.

B - It helps in deciding the dependencies of objects.

C - It stores objects states in database.

D - It stores object states in file system.

Q 5 - What AOP stands for?

A - Aspect Oriented Programming

B - Any Object Programming

C - Asset Oriented Programming

D - Asset Oriented Protocol

Q 6 - What is true about cross-cutting concerns?

A - The functions that span multiple points of an application are called cross cutting concerns.

B - Cross-cutting concerns are conceptually separate from the application's business logic.

C - Logging is one of the examples of cross cutting concerns.

D - All of the above.

Q 7 - Which are the modules of core container?

A - Beans, Core, Context, SpEL

B - Core, Context, ORM, Web

C - Core, Context, Aspects, Test

D - Bean, Core, Context, Test

Q 8 - Which are the modules of Data Access/ integration layer?

A - JDBC, ORM, OXM, JMS, Transactions

B - JDBC, ORM, OXM, JMS

C - JDBC, ORM, Web, Beans

D - JDBC, ORM, OXM, JMS

Q 9 - Which are the modules of Web layer?

A - WebSocket, Servlet, Web, Portlet

B - WebSocket, Servlet, Web-MVC, Web

C - HTML, JSP, WEB, Portlet

D - HTML, Servlet, WEB, Portlet

Q 10 - Which of the statement is not correct?

A - Core and beans modules provide the fundamental parts of the framework, including Dependency Injection feature.

B - The SpEL module provides a powerful Expression Language for querying and manipulating an object graph at runtime.

C - Aspects module provides integration with AspectJ.

D - None of the above.

Q 11 - Which of the statement is correct?

A - The JDBC module provides a JDBC-abstraction layer that removes the need to do tedious JDBC related coding.

B - The ORM module provides integration layers for popular object-relational mapping APIs, including JPA, JDO, Hibernate, and iBatis.

C - The Java Messaging Service JMS module contains features for producing and consuming messages.

D - All of the above.

Q 12 - Which of the statement is correct?

A - The AOP module provides aspect-oriented programming implementation allowing you to define method-interceptors and pointcuts to cleanly decouple code that implements functionality that should be separated.

B - The Aspects module provides integration with AspectJ - Which is again a powerful and mature aspect oriented programming AOP framework.

C - The Instrumentation module provides class instrumentation support and class loader implementations to be used in certain application servers.

D - All of the above.

Q 13 - What types of Dependency injection does spring supports?

A - Constructor based, Setter based

B - Constructor based, Setter based, Getter Based

C - Setter based, Getter based, Properties based

D - Constructor based, Setter based, Properties based

Q 14 - Which are the IoC containers in Spring?

A - BeanFactory, ApplicationContext

B - BeanFactory, ApplicationContext, IocContextFactory

C - BeanFactory, BeanContext, IocContextFactory

D - BeanFactory, ApplicationContext, BeanContext

Q 15 - Which is the correct implementation class of BeanFactory?

A - XmlBeanFactory

B - ClassPathBeanFactory

C - FileSystemBeanFactory

D - AdvancedBeanFactory

Q 16 - Which are the correct implementation classes of ApplicationContext?

A - FileSystemXmlApplicationContext, ClassPathXmlApplicationContext, WebXmlApplicationContext

B - FileSystemApplicationContext, ClassPathApplicationContext, WebApplicationContext

C - AdvancedApplicationContext, FileApplicationContext

D - FileSystemApplicationContext, ClassPathApplicationContext

Q 17 - Which of the following stands true for spring beans?

A - Spring beans are managed by the Spring IoC container.

B - Spring beans are instantiated, assembled, and otherwise managed by a Spring IoC container.

C - Spring beans are simple POJOs.

D - All of the above.

Q 18 - Which is the way to provide configuration metadata to spring?

A - XML Based configuration file.

B - Annotation based configuration.

C - Java based configuration.

D - All of the above.

Q 19 - What is bean scope?

A - Bean scope forces Spring to produce a new bean instance as per the scope defined.

B - Bean scope defines the accessibility of bean in a java class.

C - Bean scope defines the accessibility of bean in a java package.

D - Bean scope defines the accessibility of bean in a web application.

Q 20 - What is singleton scope?

A - This scopes the bean definition to a single instance per Spring IoC container.

B - This scopes the bean definition to a single instance per HTTP Request.

C - This scopes the bean definition to a single instance per HTTP Session.

D - This scopes the bean definition to a single instance per HTTP Application/ Global session.

Q 21 - What is prototype scope?

A - This scopes a single bean definition to have any number of object instances.

B - This scopes the bean definition to a single instance per HTTP Request.

C - This scopes the bean definition to a single instance per HTTP Session.

D - This scopes the bean definition to a single instance per HTTP Application/ Global session.

Q 22 - What is request scope?

A - This scopes a bean definition to an HTTP request.

B - This scopes the bean definition to Spring IoC container.

C - This scopes the bean definition to HTTP Session.

D - This scopes the bean definition HTTP Application/ Global session.

Q 23 - What is session scope?

A - This scopes a bean definition to an HTTP session.

B - This scopes the bean definition to Spring IoC container.

C - This scopes the bean definition to HTTP request.

D - This scopes the bean definition to HTTP Application/ Global session.

Q 24 - What is global-session scope?

A - This scopes a bean definition to an HTTP Application/ Global session.

B - This scopes the bean definition to Spring IoC container.

C - This scopes the bean definition to HTTP request.

D - This scopes the bean definition to HTTP Session.

Q 25 - What is default scope of bean in Spring framework?

A - singleton

B - prototype

C - request

D - session

Q 26 - How can you inject Java Collection in Spring?

A - Using list, set, map or props tag.

B - Using lit, set, map or collection tag.

C - Using list, set, props or collection tag.

D - Using list, collection, map or props tag.

Q 27 - What is true about <list> collection configuration elements?

A - This helps in wiring a list of values, allowing duplicates.

B - This helps in wiring a list of values but without any duplicates.

C - This can be used to inject a collection of name-value pairs where name and value can be of any type.

D - This can be used to inject a collection of name-value pairs where the name and value are both Strings.

Q 28 - What is true about <set> collection configuration elements?

A - This helps in wiring a list of values, allowing duplicates.

B - This helps in wiring a list of values but without any duplicates.

C - This can be used to inject a collection of name-value pairs where name and value can be of any type.

D - This can be used to inject a collection of name-value pairs where the name and value are both Strings.

Q 29 - What is true about <map> collection configuration elements?

A - This helps in wiring a list of values, allowing duplicates.

B - This helps in wiring a list of values but without any duplicates.

C - This can be used to inject a collection of name-value pairs where name and value can be of any type.

D - This tag is not supported.

Q 30 - What is true about <props> collection configuration elements?

A - This helps in wiring a list of values, allowing duplicates.

B - This helps in wiring a list of values but without any duplicates.

C - This can be used to inject a collection of name-value pairs where name and value can be of any type.

D - This can be used to inject a collection of name-value pairs where the name and value are both Strings.

Q 31 - What is bean autowiring?

A - Autowiring lets Spring resolve collaborators *otherbeans* for your bean by inspecting the contents of the BeanFactory without using <constructor-arg> and <property> elements.

B - Autowiring injects values in spring beans.

C - Autowiring injects one bean into another.

D - Autowiring helps in wiring a list of values, allowing duplicates.

Q 32 - Which are the different modes of autowiring?

A - no, byName, byType, constructor, autodetect

B - no, byName, byType, constructor, autocorrect

C - byName, byContent, constructor, autodetect

D - byName, byContent, setter, autodetect

Q 33 - What is no mode of autowiring?

A - Default setting which means no autowiring and you should use explicit bean reference for wiring.

B - Autowiring by property name.

C - Spring first tries to wire using autowire by constructor, if it does not work, Spring tries to autowire by byType.

D - Similar to byType, but type applies to constructor arguments.

Q 34 - What is byName mode of autowiring?

A - Default setting which means no autowiring and you should use explicit bean reference for wiring.

B - Autowiring by property name. Spring tries to match and wire its properties with the beans defined by the same names in the configuration file.

C - Spring first tries to wire using autowire by constructor, if it does not work, Spring tries to autowire by byType.

D - Similar to byType, but type applies to constructor arguments.

Q 35 - What is byType mode of autowiring?

A - Default setting which meas no autowiring and you should use explicit bean reference for wiring.

B - Autowiring by property name. Spring tries to match and wire its properties with the beans defined by the same names in the configuration file.

C - Spring first tries to wire using autowire by constructor, if it does not work, Spring tries to autowire by byType.

D - Autowiring by property type. Spring tries to match and wire a property if its type matches with exactly one of the beans name in configuration file.

Q 36 - What is constructor mode of autowiring?

A - Autowiring by property name. Spring tries to match and wire its properties with the beans defined by the same names in the configuration file.

B - Spring first tries to wire using autowire by constructor, if it does not work, Spring tries to autowire by byType.

C - Autowiring by property type. Spring tries to match and wire a property if its type matches with exactly one of the beans name in configuration file.

D - Similar to byType, but type applies to constructor arguments. If there is not exactly one bean of the constructor argument type in the container, a fatal error is raised.

Q 37 - What is autodetect mode of autowiring?

A - Similar to byType, but type applies to constructor arguments. If there is not exactly one bean of the constructor argument type in the container, a fatal error is raised.

B - Autowiring by property name. Spring tries to match and wire its properties with the beans defined by the same names in the configuration file.

C - Spring first tries to wire using autowire by constructor, if it does not work, Spring tries to autowire by byType.

D - Autowiring by property type. Spring tries to match and wire a property if its type matches with exactly one of the beans name in configuration file.

Q 38

Can you inject null and empty string values in Spring?

A - Yes

B - No

Q 39 - How do you turn on annotation wiring?

A - Add <annotation-context:config /> to bean configuration.

B - Add <annotation-config /> to bean configuration.

C - Add <annotation-context-config /> to bean configuration.

D - Add <context:annotation-config/> to bean configuration.

Q 40 - What does @Required annotation mean?

A - This annotation indicates that bean property must be populated by the user.

B - This annotation indicates that bean property is required while saving the bean data to database.

C - This annotation simply indicates that the affected bean property must be populated at configuration time, through an explicit property value in a bean definition or through autowiring.

D - This annotation indicates that bean property is required while serializing the bean.

Q 41 - What is true about @Autowired annotation?

A - The @Autowired annotation can be used to autowire bean on the setter method.

B - This annotation provides more fine-grained control over where and how autowiring should be accomplished.

C - The @Autowired annotation can be used to autowire bean on the methods with arbitrary

names and/or multiple arguments.

D - All of above.

Q 42 - What is ContextRefreshedEvent event?

A - This event is published when the Servlet Context is either initialized or refreshed.

B - This event is published when the HTTP Request is received.

C - This event is published when the HTTP Response is returned.

D - This event is published when the ApplicationContext is either initialized or refreshed.

Q 43 - What is ContextStartedEvent event?

A - This event is published when the Servlet Context is either initialized or refreshed.

B - This event is published when the HTTP Request is received.

C - This event is published when the ApplicationContext is started using the start method on the ConfigurableApplicationContext interface.

D - This event is published when the HTTP Response is returned.

Q 44 - What is ContextStoppedEvent event?

A - This event is published when the Servlet Context is either initialized or refreshed.

B - This event is published when the ApplicationContext is stopped using the stop method on the ConfigurableApplicationContext interface.

C - This event is published when the HTTP Request is received.

D - This event is published when the HTTP Response is returned.

Q 45 - What is ContextClosedEvent event?

A - This event is published when the Servlet Context is either initialized or refreshed.

B - This event is published when the HTTP Request is received.

C - This event is published when the HTTP Response is returned.

D - This event is published when the ApplicationContext is closed using the close method on the ConfigurableApplicationContext interface.

Q 46 - What is RequestHandledEvent:event?

A - This event is published when the Servlet Context is either initialized or refreshed.

B - This event is published when the HTTP Request is received.

C - This event is published when the HTTP session is initialized or refreshed.

D - This event is published when the HTTP Request is serviced.

Q 47 - What is aspect?

A - Aspect is a way to do the dependency injection.

B - A module which has a set of APIs providing cross-cutting requirements.

C - Aspect is used to log information of application.

D - Aspect represents properties of spring based application.

Q 48 - What is Join point?

A - This represents a point in your application which joins two objects.

B - This represents a point in your object where you join values.

C - This represents a point in your object where you join injected values.

D - This represents a point in your application where you can plug-in AOP aspect.

Q 49 - What is Advice?

A - This is the way to instruct object to behave in certain manner.

B - This is used to inject values in objects.

C - This is the actual action to be taken either before or after the method execution.

D - This is not invoked during program execution by Spring AOP framework.

Q 50 - What is Pointcut?

A - This represents a point in your application where you can plug-in AOP aspect.

B - This is a set of one or more joinpoints where an advice should be executed.

C - This is used to inject values in objects.

D - This is invoked during program execution by Spring AOP framework.

ANSWER SHEET

Question Number	Answer Key
1	A
2	D
3	A
4	A
5	A
6	A
7	A
8	A

9	A
10	D
11	D
12	D
13	A
14	A
15	A
16	A
17	D
18	D
19	A
20	A
21	A
22	A
23	A
24	A
25	A
26	A
27	A
28	B
29	C
30	D
31	A
32	A
33	A
34	B
35	D
36	D
37	C
38	A
39	D
40	C
41	B
42	D
43	C

44	B
45	D
46	C
47	B
48	D
49	C
50	B

Loading [MathJax]/jax/output/HTML-CSS/jax.js