https://icompass.fs.capgemini.com/iCompass/images/icompass_logo.png

Java\_Spring\_Topics\_A3 To A5\_Cloud1

https://icompass.fs.capgemini.com/iCompass/images/capgemini.png

**Section:**

* **1**

**Time Left :** **00 : 35**

**List View**



*Active     Attempted     Not Attempted*

**Q1**

In DAO, The implementation of an application object requires only hibernate session factory.  
This can be provided as a simple bean reference from the spring application context.   
Select the valid bean implementation for the above scenario:

*Single Choice - Select one correct answer from the options list.*

<beans>  
   <bean id=”emplDao” class=”emp.empDao”>  
   <property name=”sessionfactory” ref=”mysessionfactory”>  
   </bean>  
   </beans>

<beans id=”emplDao” class=”emp.empDao”>  
   <property name=”sessionfactory” ref=”mysessionfactory”>  
   </beans>

<beans>  
  < bean id=”emplDao” class=”emp.empDao”>  
   <property name=”\*” ref=”mysessionfactory”>  
 </bean>  
 </beans>

<beans>  
     <property name=”\*” ref=”mysessionfactory”>  
 </bean>  
 </beans>

**Q2**

Annotation @Autowire can be used in Constructors and setter methods only?

*Single Choice - Select one correct answer from the options list.*

TRUE

FALSE

**Q3**

What is the use of JdbcTemplate in spring?

*Single Choice - Select one correct answer from the options list.*

Class that manages connection to databases

Eliminates boiler plate code

Handles SQL Exception properly

All of the above

**Q4**

Consider the following code   
. <tx:advice id="txAdvice" transaction-manager="txManager">   
<tx:attributes>   
  <tx:method name="\*" read-only="true" rollback-for="NoEmployeeFound"/>   
</tx:attributes>   
</tx:advice>  
Based on the sample code above, the rollback-for attribute within the <tx:method> element causes Spring to rollback a transaction if a(n):

*Single Choice - Select one correct answer from the options list.*

Checked exception of the class "NoEmployeeFound" or any unchecked exception or error is thrown.

checked exception of the class "NoEmployeeFound" is thrown.

throwable containing the message "NoEmployeeFound" is thrown.

unchecked exception of the class "NoEmployeeFound" or any checked exception or error is thrown.

**Q5**

Which of the following is not the types of Advice?

*Single Choice - Select one correct answer from the options list.*

Before advice.

After returning advice.

After completion advice.

After advice.

**Q6**

Statement A:  
The TransactionTemplate adopts the same approach as other spring templates such as JdbcTemplate.  
Statement B:  
TransactionTemplate is not threadsafe

*Single Choice - Select one correct answer from the options list.*

Both Statement A and Statement B are false.

Both Statement A and Statement B are true.

Statement A is true and Statement B is false .

Statement A is false and Statement B is true

**Q7**

Which build tools are supported by  
Spring Boot framework out of box

*Multiple Choice - This may have multiple correct answers. Select required answer(s) from the options list.*

Maven

Gradle

Ant

Make

**Q8**

Join point is always a method in Spring AOP.  
    State true or false.

*Single Choice - Select one correct answer from the options list.*

True.

False.

**Q9**

What is data-source in spring?

*Single Choice - Select one correct answer from the options list.*

Means to provide connection details to the database

Input file reader from which you can read data

Both A and B

None of the above

**Q10**

Which of the following annotations can be used on fields, parameters or custom annotations?

*Single Choice - Select one correct answer from the options list.*

@autowired

@resource.

@Qualifier.

@component.

**Q11**

How do you enable spring to scan for declaration of MVC layer beans in spring?

*Single Choice - Select one correct answer from the options list.*

<mvc:annotation-init/>

<mvc.annotation-load/>

<mvc:annotation-driven/>

None of the above

**Q12**

Following property is set for what reason  
spring.autoconfigure.exclude=  
my.xyz.SomeAutoConfig

*Single Choice - Select one correct answer from the options list.*

To exclude unnecessary or misconfigured   
Auto configurations via  
properties

Reconfiguring   
AutoConfiguration via  
properties

Overriding conditions

For completely  
custom   
Autoconfiguration

**Q13**

Spring's support for annotations such as @Autowired and @Resource, provides dependency injection for private/protected fields, setter methods, and configuration methods.  
State true or false.

*Single Choice - Select one correct answer from the options list.*

True.

False.

**Q14**

Does spring handle Multipart request?

*Multiple Choice - This may have multiple correct answers. Select required answer(s) from the options list.*

No by default

Yes, if we configure Multipart resolver with commons library in classpath

There is no such thing as Multipart request

None of the above

**Q15**

Which dependency need to be  
mentioned to enable Spring MVC   
module in an application

*Single Choice - Select one correct answer from the options list.*

spring-boot-starter-data-jpa

spring-boot-starter-web

spring-boot-starter-test

spring-boot-starter-tomcat

**Q16**

Which of the following tag is used navigate through the collection?

*Single Choice - Select one correct answer from the options list.*

<util:list/>

<util:constant/>

<util:property-path/>

<util:properties/>

**Q17**

Which of the following are the benefits of Aspect Oriented Programming?

*Single Choice - Select one correct answer from the options list.*

More Code reuse

Late Binding of design decisions.

Higher modularization

All the above

**Q18**

How do you read request parameter values into spring controller methods?

*Single Choice - Select one correct answer from the options list.*

You cannot bind them at all

You cannot bind them and have to manually do request.getParameter()

You can do either request.getParameter inside method or @RequestParam in method args

None of the above

**Q19**

What are the ways of resolving constructor arguments for the following class in Xml?

class MyClass {

           public MyClass(int firstval, int secondVal) {

          }

}

*Multiple Choice - This may have multiple correct answers. Select required answer(s) from the options list.*

Specify arguments in the required order in the bean defination

Specify the index attribute for each constructor argument

Specify the type attribute for each constructor argument

user the @ConstructorProperties argument and name attribute for each constructor arg

**Q20**

In which of the following jar, the integration testing classes are packaged?

*Single Choice - Select one correct answer from the options list.*

spring-core.jar

spring-integration-test.jar

spring-beans.jar

spring-support.jar

**Q21**

In Unit Testing ,\_\_\_\_\_\_\_ package contains a set of Portlet API mock objects, targeted at the usage with Spring portlet MVC framework

*Single Choice - Select one correct answer from the options list.*

org.springframework.mock.web.portlet

org.springframework.mock.web

org.springframework.jndi.

org. springframework.unittesting.

**Q22**

The Java Bean Property is automatically populated by \_\_\_\_\_\_\_\_\_

*Single Choice - Select one correct answer from the options list.*

DelegatingResponseProcessor.

ResponseProcessor

DelegatingRequestProcessor

RequestProcessor.

**Q23**

Which feature is not supported by Spring framework

*Single Choice - Select one correct answer from the options list.*

IOC

Interface Injection

JDBC Transaction

JDBC Exception Handling

**Q24**

Choose the ORM tools that are supported by spring Framework : (Choose all that apply)

*Multiple Choice - This may have multiple correct answers. Select required answer(s) from the options list.*

Hibernate.

Ibatis.

Java Persistence API.

Top Link.

**Q25**

Spring has its own Model View Controller (MVC), but it can be integrated with other web frameworks like Struts. State true or false

*Single Choice - Select one correct answer from the options list.*

TRUE

False.

**Q26**

Which of the following method is used to return a TransactionStatus object that depends on a TransactionDefinition parameter?

*Single Choice - Select one correct answer from the options list.*

getTransaction()

setTransaction()

isNewTransaction()

setRollbackOnly()

**Q27**

What is the use of <mvc:annotation-driven/> in spring?

*Single Choice - Select one correct answer from the options list.*

It configures the basic setup for a MVC application

It just let spring know application is using Annotations

There is no such xml element in Spring

None of the above

**Q28**

Which of the following is valid Configuration for JNDI instead of Datasource in Application Context.XML?

*Single Choice - Select one correct answer from the options list.*

<bean id=dataSource class=org.springframework.jndi.JndiObjectFactoryBean>  
  <property name="jndiName">  
    <value>java:comp/env/jdbc/appfuse</value>  
  </property>  
</bean>

<bean id="dataSource" class=org.springframework.jndi.JndiObjectFactoryBean>  
  <property name="jndiName">  
    <value>java:comp/env/jdbc/appfuse  
  </property>  
</bean>

<bean id="dataSource" class=org.springframework.JndiObjectFactoryBean.>  
  <property name="jndiName">  
    <value>java:comp/env/jdbc/appfuse</value>  
  </property>  
</bean>

<bean id="dataSource" class=org.springframework.jndi.JndiObjectFactoryBean>  
  <property name="jndiName">  
    <value>java:comp/env/jdbc/appfuse</value>  
  </property>  
</bean>

**Q29**

What is the purpose of init-method and destroy- method in Spring bean?

*Single Choice - Select one correct answer from the options list.*

There are no such methods in spring bean

Init-method helps with performing custom initialization while destroy method helps in cleaning up any resources once the context is destroyed.

Calling destroy-method crashes the spring container while init-method restarts the spring container

None of the above

**Q30**

package com.acme.hr;  
  
public class Employee {  
private String employeeId = "Default";  
public void setEmployeeId(String employeeId) {  
  this.employeeId = employeeId;  
}  
}  
Based on the sample code above, what do you use to set the employeeId attribute to null?

*Single Choice - Select one correct answer from the options list.*

<bean id="employee" class="**com.acme.hr.**Employee">  
<property name="employeeId">  
  <value>null</value>  
</property>  
</bean>

<bean id="employee" class="**com.acme.hr.**Employee">  
<property name="employeeId"/>  
</bean>

<bean id="employee" class="com.hr.acme.Employee">  
<property name="employeeId">  
  <value><null/></value>  
</property>  
</bean>

<bean id="employee" class="**com.acme.hr**.Employee">  
<property name="employeeId" value="null"/>  
</bean>

Spring

123456789101112131415161718192021222324252627282930

1 out of 30 Submit Assessment

