

\*\*\*\*\* JAVA MCQ 3 \*\*\*\*\*

1) Refer the incomplete code given below:

Refer the incomplete code given below:

```
<beans.....>
```

```
<!--Assume DataSource, EntityManagerFactory configurations are provided....-->
```

Line1

```
<bean id="transcationManager"
```

```
Class="org.springframework.orm.jpa.Jpa.TransactionManager">
```

```
<property name="entityManagerFactory" ref="cs._entityManagerFactory"!>
```

```
</bean>
```

```
</beans>
```

Choose from below the valid option to enable declaration of Transaction Management using annotations.

**Ans:- <tx:annotation-driven transaction-manager-ref="transactionManager"/>**

2) Refer the below Entity class

```
@Entity
```

```
public class EmployeeEntity
```

```
{
```

```
@Id
```

```
private int employeeId;
```

```
private String employeeName;
```

```
private Double salary;
```

```
//Getters and Setters are coded
```

```
}
```

Choose from below a valid option for spring JPA data method to Fetch the Employee details who gets salary within given range in the descending order of salary.

**Ans:- A- List<Employee Entity> findBySalaryBetweenOrderBySalaryDesc(Double param1, Double param2))**

3) Which of the following are with respect to Spring transaction propagation types?

i) propagation type Propagation.REQUIRED starts a new transaction, if a method is provided from another method and a transaction does not exists.

ii) propagation type Propagation REQUIRED\_NEW starts a new transaction, if a method is invoked from another method and a t Choose the most appropriate option

**Ans:-A- Both i and ii**

4) Package com.accenture.lkm;

Line 1

```
class Customer{
```

```
private int customerId;
```

Line 2

```
private Address address;  
}
```

Line 3

```
class Address{  
public Address(){  
System.out.println("address") } }  
Line4_
```

```
@componentScan(basePackages="com")  
Public class springConfig
```

```
}  
Public class Tester{ .....  
Ans:-A- @Component,@Autowired,@Component,@Configuration,AnnotationConfigApplicationContext
```

```
5) @Entity class BookEntity{ @Id private int bookId; private String authorName; private String publisherType;  
//getter and setters }  
( QUESTION NOT FOUND)
```

6) @Configuration

```
public class AppConfig {  
@Bean("address")  
public Address creatAddress()  
{  
Address address= new Address();  
return Address;}}
```

```
Ans:- ApplicationContext ctx=new AnnotationConfigApplicationContext(AppConfig.class);  
Address address=(Address) ctx.getBean("address"); }}
```

7) Refer the following JSP pages

Header.jsp:

```
<% @ page language = "java" contentType= "text/html; charset = ISO-8859-1"  
pageEncoding = "ISO-8859-1" %>  
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" HTTP://www.w3.org/TR/html4/loose.dtd>  
<html><head>  
<meta http-equiv = "Content-Type" content = "text/html; charset = ISO-8859-1">  
<title> insert title here </title></head>  
<body>  
<h2> <font color = "blue"> This is content of Header </font> </h2>  
</body></html>
```

Footer.jsp

```
<% @ page language = "java" contentType= "text/html; charset = ISO-8859-1"
```

```

pageEncoding = "ISO-8859-1" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" HTTP://www.w3.org/TR/html4/loose.dtd>
<html><head>
<meta http-equiv = "Content-Type" content = "text/html; charset = ISO-8859-1">
<title> insert title here </title></head>
<body>
<h2> <font color = "Green"> This is content of Header </font> </h2>
</body></html>

```

```

Home.jsp
<% @ page language = "java" contentType = "text/html; charset = ISO-8859-1"
pageEncoding = "ISO-8859-1" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" HTTP://www.w3.org/TR/html4/loose.dtd>
<html><head>
<meta http-equiv = "Content-Type" content = "text/html; charset = ISO-8859-1">
<title> insert title here </title></head>
<body>
<%@ include file="Header.jsp" %>
<font color="red">
<h2> This is the main page content</h2>
<% int i=100; %>
<%=i%>
</font>
<%@ include file="footer.jsp" %>
</body></html>

```

How many servlets would be created while executing home.jsp?

**Ans:- A- 1**

```

8) Class Library{
private Book book;
Public Library(Book book){
this.book=book; } }

```

```

Class Book{
private string bookName;
public Book(String bookName){
this.bookName= bookName; } }

```

Choose from below a valid option to define Book as a inner bean of library

```

Ans:-B- <bean id="library" class="com.accenture.lkm.library">
<constructor-arg>
<bean class="com.accenture.lkmBook">
<constructor-arg value="book1">< constructor-arg>

```



```
</bean>
</constructor-arg>
</bean>
```

9) login.html;

```
<html>
<body>
<h2>Login Page</h2>
<form action="sessionServlet1">
<table>
<tr><td>UserName</td><td>.....
<tr><td>Password</td><td>.....
</table>
.....John, admin.
```

**QUESTION NOT FOUND**

10) Refer the below code for validating the user.

Login.jsp:

```
<html><body>
<h2>Login Page</h2>
<form action="login" method="POST">
<table>
<tr><td>UserName</td><td><input type="text" name=".....
<tr><td>Password</td><td><input type="password" name=".....
</table>
<input tupe="submit" value="Login"/></form></body></html>
```

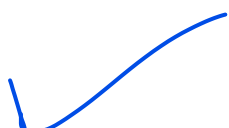
LoginServlet.java:

```
Package com.accenture.lkm;Import java.io.*; Import javax.servlet.*; Import
javax.servlet.http.*;
Public class LoginServlet extends HttpServlet{
Private static final long.....
```

```
.....
.....
```

Ans:- A- <servlet>

```
<description></description>
<display-name>LoginServlet</display-name>
<servlet-name>LoginServlet</servlet-name>
<servlet-class>com.accenture.lkm.LoginServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>LoginServlet</servlet-name>
```



<url-pattern>login</url-pattern>  
<servlet-mapping>

11) Consider the code given below:

```
Package com.accenture.lkm:  
@Component("address")  
Public class Address{  
@Value("Hyderabad- Telangana")  
Private String addressLine1;  
//Getters and setters are already coded  
}  
Package com.accenture.lkm:  
@Component("employee")  
Public class Employee{  
@Autowired Private Address address;  
//Getters and setters are already coded  
}  
Public class UITester{ .....  
System.out.println("Employee Address:"+employee.getAddress().getAddressLine1());  
} } ("Bangalore-Maharashtra")
```

Ans:- B- Employee Address: Bangalore-Maharashtra

```
12) @Entity  
class BookEntity  
{  
@Id  
private int bookId;  
private String title;  
private String authorName;  
private String publisher Type;  
// getters and setters  
}
```

Choose from below the valid option to retrieve the books based on given publisher type by using Query Method Approach

Ans:- A- findByPublisherType(String pType)

```
13) @Entity  
public class CustomerEntity{  
@Id  
private int cid;  
private String cname;  
private double credit;
```

// getter and setter methods

Choose the valid implementation of the delete query to delete customers who has credit less than a particular value.

Ans:- A- @Modifying

@Query("delete from CustomerEntity k where k.credit<cpoint)int  
delete(@Param('cpoint') double credit);

14) Code for creating employee=null @Transactional(value = "txManager", rollbackFor = NullPointerException.class)

Ans:- A. @Transaction("value ="txManager",rollback=NullPointerException.class)

15) Refer the JSP code below and predict the output

```
<%! int bookId = 1; int publish Year = 2000; public int incrementBookId(int bookId) {return bookId++; } %>  
<% publish Year++; bookId++; %>  
Book Id: <% = bookId %>
```

Publish Year: <% = publish Year %>

Predict the Output.

Ans:- Book Id: 2 publish Year: 2001

16) Which lifecycle method of jsp is invoked by the container to perform the actual task also the method is invoked by the container each ....

Ans:- B- \_jspService()

17) Farhan wants to write a code to add 2 number and return the result. He also wanted to place the code outside the service method. Help him to select the appropriate JSP.....element

Ans:- B- <%! public int add(int num1, int num2){ written num1+num2; } %>  
return

18) From the following identify the correct hierarchy of servlets assuming you are required to create a registration servlet to process registration request.

Ans:- A-

RegistrationServlet.java extends HTTPServlet.

HttpServlet extends GenericServlet.

GenericServlet implements Servlets

19) When you want to remove the user data from session, what all the various options available in JSP?

A-To delete the session attribute we can call public void removeAttribute(String name)

B-by invoking public void invalidate()

C-in web.xml file use <session-timeout> 15 </session-timeout>

D)Perform log out and delete the session data

Ans:- A,B,C

A and B and D

**20)** Which is an abstract class which provides the basic implementation of servlet interface except service() method?

**Ans:- C- GenericServlet**

**21)** Refer the code given below:

```
package com.model;  
@component("prodObject")  
@Profile("myProdProfile")  
public class Product {  
    private int productId;  
    private String productName;  
}
```

```
package com.model;  
@component("orderObject")  
@Profile("myOrderProfile")  
class Order {  
    private int orderId;  
    private String orderDetail; }
```

choose from the below the valid option to activate ....

**Ans:-A- @ActiveProfiles(profiles="myorderProfile")**

**22)** Refer the code given below

```
@Entity  
public class EmployeeEntity {  
    @Id  
    private int emplId;  
    private String name;  
    private double salary;  
    //getter and setter methods }
```

Choose the valid implementation of the Update query to update employee salary, using Spring JPA Data.

**Ans:- A- @Modifying @Query("update EmployeeEntity k set k.salary='newSal')Int  
updateSalary(@Param("newSal")Double salary);**

**23)** From the following identify TWO INCORRECT statements about ServletConfig objects.

- 1) It is not created for each servlet during servlet initialization.
- 2) It will be available to all servlets of an application.
- 3) Its lifetime is until the servlet class object is destroyed.
- 4) One object per servlet class

**Ans:- A- i and ii only** 1 and 3 are incorrect

**24)** Raghav must pre populate doctor's names from doctors table in a drop down list when patient is booking appointment online..He has written required method implementation in DAO and service layer.

Raghav added following query in orm.xml file select p.pName,p.appointmentDate from patients p Select the correct method declaration in DAO interface. Refer the patients class and patients table.

Ans:-

```
@Query(name="getPatientData")  
List<string>getPatientsData();
```

Maybe List<Patients> findAllDoctors();

25) Raghav has added a method to update the details of the patients as below public interface Patients DAO {  
@Query(name="updateQuery1") int updateAppointmentSlot(Patients patients, String new Appointment(Time)); }  
But he getting exception as "DML operation is not supported".What went wrong?

Ans:- @Modifying is not used on the method output should return the string data which is the updated appointment time.

26) Raghav added the following methods in Service and DAO implementation classes to add patient's information in db. //service implementation @transactional(propagation=Propagation.REQUIRED) Integer  
addPatientInService(Patients patient) { //calling DAO method } //DAO implementation  
@transactional(propagation=Propagation.REQUIRES\_NEW) Integer addPatientInService(Patients patient) {  
//implementation } what is the behaviour of the addPatientInDAO() method?

Ans:- addPatientInDAO() method always executes in a new transaction rather than the interface started in service layer of addPatientInService() method.

27) Which of the following is/are correct statement(s) with respect to spring test?  
(i)@DirtiesContext is used to mark the context dirty so that it can be closed  
(ii)@ContextConfiguration can only point to Spring XML configuration but not java configuration  
Choose the most appropriate option.

Ans:- Only i ii is correct

28) Refer the incomplete code given below:

```
@Component class Customer(  
.....  
private Cart cart;  
}
```

Assume instance of the Cart is managed by Spring Container by the name "cartObj" and all the classes and configuration are coded properly. Choose from below a valid SpEL to inject the cart object in Customer bean;

Ans:- A- @Value("#{cartObj}")

29) /springormdemos"/>

```
<property name ="username" value....;  
<property name="password" value....;  
</bean>  
<bean>
```

```
Id-"cst_entityManagerFactory" class=*org.springframework.com...  
/// dataSource , JpaVendorAdapter"
```



```
Class="org.springframework.orm.jpa.vendor.hibernate.jpavendoradapter
<property name="showSql"....
<property name="generate.. Value="org.hibernate.dialect.MySQL5Dialect"/>
.....
```

Ans:-A= DriverManagerDataSource, LocalContainerEntityManagerFactoryBean,@repository,@Auto f

30) Mr. Mark has been asked to design a JSP page. He should declare variables to store "...", ".....", "quantity" and "price". Total price must be calculated and displayed using an expression tag. From the following identify CORRECT script to be added in the JSP body section. Choose one

```
Ans:- A) <%! Int ItemId=1; String UserName="N-95 Mask";Int quantity=10; Int price=50; Int
totalPrice=quantity*price; %> <%=totalprice %>
<%! Int ItemId=1; String UserName="N-95 Mask";Int quantity=10;
Int price=50; Int
totalPrice=quantity*price; %>
<%=totalPrice %>
```

31) Which among the following is used to load the Spring xml configuration for test cases using Spring Test? Choose the most appropriate option

Ans:- @ContextConfiguration(locations="/com/resources/my\_springbean.xml ")@Test

32) Refer the incomplete code below

```
package com.accenture.ikm;
public class job{
private int jobid;
//getter and setter }
public class person{
private String[] skills;
Line1 privatejob job;
Person(String[]Skills){ this.skills=skills; } }
```

Ans:- A- @Value("#myjob"), @Configuration, @Bean, @Bean(name="myjob")

33) Refer the code given below:@Entityclass BookEntity{@idprivate int bookid; private String title; private String authorName; private String publisherType; }

Ans:- findByPublisherType(String p Type)

35) Refer the code below:Web.xml

```
<web-app>
<servlet>
<servlet-name>Book</servlet-name>
<servlet-class>com.accenture.BookServlet</servlet-class>
<load-on-startup>0</load-on-startup>
</servlet>
<servlet>
<servlet-name>Magazine</servlet-name>
<servlet-class>com.accenture.MagazineServlet</servlet-class>
```

```
<load-on-startup>1</load-on-startup>
</servlet>
</web-app>
```

**Ans:- Book Servlet**

**36)** Choose from below valid options which provides complete support for creation of JPA specific EntityManagerFactory and Transaction Management.

**Ans:- B- LocalContainerEntityManagerFactoryBean**

**37)** Mr. John has been assigned the task of creating a servlet filter to log IP addresses of the computers from which the request originates. Help him from the following options to identify the correct order of tags meant for servlet filters in web.xml file.

**Ans:- A- <web-app>**

```
.....
<filter>
<filter-name>...</filter-name>
<filter-class>...</filter-class>
</filter>
<filter-mapping>
<filter-name>...</filter-name>
<url-pattern>...</url-pattern>
</filter-mapping>
.....
</web-app>
```

**38)** Refer the code given below @Component @Profile("myProfile") class Product{} Choose from below a valid option to activate the profile "myprofile"

**Ans:- B- System.setProperty("spring.profiles.active", "myProfile");**

**39)** John has created a properties file below:

Config.properties

customerName: James

CustomerId: 1009090

Choose from below a valid option to inject values in Customer bean from config.properties file

**Ans:-A- @Component**

```
public class Customer
{
@Value("${customerID}")
Private int customerId;
@Value("${customerName}")
private String customerName;
}
```

**40)** Which of the following statements are correct?

1. JVM supports IOC
2. Spring Container manages the instance of the beans mentioned in the configuration and provides them back to code when needed.

**Ans:- Only ii**

**41)** Which of the following is correct with respect to spring?

- i) @Configuration is used to declare a XML file as a Spring configuration
- ii) @ComponentScan is used to scan stereotype annotation.

**Ans:- Only ii**

**42)** Which of the following are stereotype annotations? [Choose 2]

**Ans:- A & B- @component @service**

**43)** Komal's task is to create a customer report on daily..... everyday when she generates a report help her to identify..... task.

i) `<% java.util.Date date =new java.util.Date();out.print(date); %>`

ii) `<% java.util.Date date =new java.util.Date(); %> <%=date;%>`

iii) `<% java.util.Date date =new java.util.Date(); %> <%date%>`

**Ans:- i and iii** all the above

**44)** Choose from below the correct statement about @Context annotation

- A) It can be placed at classes level only
- B) it is used to close and load the application context.....
- C) if no classMode is specified, the default mode BEFORE\_CLASS is used
- D) It is used to refer custom location of the spring configuration

**Ans:- B- it is used to close and load the application context....**

**45)** Refer the incomplete code given below:

`<beans>`

`<!--Assume rest of the configuration is written-->`

`<tx:annotation-driven transaction-manager="txManager"/>`

`<bean id="txManager"class="org.springframework.orm.jpa.jpa">`

`<property name="entityManagerFactory"ref="cst_entityManagerFactory">`

`</bean>`

`</beans>`

Line1

`public class EmployeeDAOImpl implements EmployeeDAO{`

Line2

`private EntityManager entityManager;`

`_____` Line3

```
public Integer addEmployee(EmployeeBean employeeBean) Integer employeeID =0;
return employeeID;}
```

Ans:- C-@Repository,@PersistenceContext,@Transactional(value="txManager ")

46) Miss Ashrey has been asked to add headers and footers while designing login page and homepage in jsp. from the following options help her to identify the right tags in JSP so that headers and footers are consistent in web pages. Moreover she has to display username captured in the login page in headers across at pages headers and footers ate html pages named Header.jsp and footer.jsp

```
Ans:-A- <jsp include page="Header.jsp">
<jsp param value="userName" name="{name}"/>
</jsp include>
<jsp include page="Footer.jsp">
</jsp include>
```

```
<jsp:include page="Header.jsp">
<jsp:param value="userName" name="name"/>
</jsp:include>
<jsp:include page="Footer.jsp">
</jsp:include>
```

47) Refer the incomplete code given below:

```
<beans.....>
<!--Assume DataSource, EntityManagerFactory configurations are
provided....-->
Line1
<bean id="transcationManager"
Class="org.springframework.orm.jpa.Jpa.TransactionManager">
<property name="entityManagerFactory" ref="cs._entityManagerFactory"!>
</bean>
</beans>
```

Choose from below the valid option to enable declaration of Transaction Management using annotations

Ans:- <tx:annotation-driven transaction-manager-ref="transactionManager"/>

48) When Servlet destroy() method will be invoked by servlet Container?

- Ans:-
- A) When you make any changes in the servlet class and save
  - C) When you stop the server, destroy method of all existing servlet objects will be invoked.

49) Choose from below the CORRECT statement about Autowiring.

Ans:-C- Autowiring requires setters or constructors in bean classes

50) Consider a web application is created with a welcome index.html mapped with a Servlet class to process the response. Refer the code below:

index.html

.....

```
<form action="LanguagesServlet" method = "POST">
<input type="checkbox" name="language" value="english"/> English
```

```

<input type="checkbox" name="language" value="hindi"/> Hindi
<input type="checkbox" name="language" value="tamil"/> Tamil
<input type="checkbox" name="language" value="malayalam"/> Malayalam
<input type="checkbox" name="language" value="telegu"/> Telegu
<input type="checkbox" name="language" value="kannada"/> Kannada
<input type="checkbox" name="language" value="french"/> French
<input type="checkbox" name="language" value="spanish"/> Spanish@
<input type="submit" value="Select. Languages"/>
</form>

```

.....Not Visible.....

**Ans:-** req.getParameterValues("languages")

**51)** Choose from the below the INCORRECT statement about JSR-300 annotation

**Anfs:-** C- JSR-330 annotations can not be used along with Spring framework Annotations

**53)** Peter wants a method to add 2 numbers and return the result. He also wanted to place it inside the service method. Help him to select the appropriate JSP scripting.

**Ans:-** <%! public int add(int num1, int num2) {return num1 + num2; } %>

**54)** <%@page language="java" contentType = "text/html;charset=ISO-8859-1"

pageEncoding = "ISO-8859-1"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Sample</title>

</head>

<body>

<%--Add JSP Script here-->

</body>

</html>

Miss Linda has been asked to design a JSP page to display company name, project name, location using JSP scripting elements such as expression and scriptlet tags.

**Ans:-A-**

<%

String companyName = "Accenture"

;String projectName = "XYZ";

String location = "MDC";

%>

<%= companyName %>

<%

String companyName = "Accenture";

String projectName = "XYZ";

String location = "MDC";

%>

<h1> <%= companyName %> </h1>

<h2> <%= projectName %> </h2>

<h3> <%= location %> </h3>

<%= projectName %>  
<%= location %>

55) Package com.accenture.lkm;  
@Component  
Public class Customer {  
//Line-X  
Private Cart cart;  
Private Integer customerId; Private String customerName;  
//Line-Y  
Public void setCart(Cart cart){This.cart=cart;  
}  
//Default Constructor and  
// Getters and Setters for other properties are coded

Assume Cart class is coded and in a spring manager bean by the name “cartObject” and component scan is done for the package com.accenture.lkm in spring configuration file. Choose from below a valid option/options so that “cartObject” can be injected into the Customer bean. Choose two

Ans:-  
Use @Autowired at LineX  
Use @Autowired at LineY

56) Miss Monica has given login request to EventmanagementApp. The app displays a JSP page to prompt her to enter the username and password. Identify the phases of JSP page it goes through until the response is generated in right order

Ans:- Translation, Compilation, Instantiation, Service and Destroy (pdf ANs)  
Translation ,Compilation, Class loading, Instantiation, Initialization, Request Processing, Destruction (google ans)      Translation, compilation, class loading, instantiation, initialization, request processing, destruction

57) Consider a web application is created with a welcome file index.html mapped with a servlet class Toprocess the response. Refer the code below:  
Ans:- B

58) Sadana wants to share the common database data throughout the application. Help her to identify the appropriate code to achieve this task.

Ans:- C)  
<%  
String userName=(String)application.getAttribute(“dbname”);  
%>

59) Refer the incomplete code given below:  
package com.accenture.lkm;  
Class House{

```

Private String houseNum;
Line1
Line2
private Room room;
//setter and getter
}
class Room{ private int roomNum;
//setter and getter }

```

**Ans: A- @Autowired, @Qualifier("room1"), com**

**60) Select correct options to refer a Spring configuration file inside a package? Choose 2**

**Ans:-**

**A- ApplicationContext applicationContext=new**

**ClassPathXmlApplicationContext("com/accenture/lkm/resource/my\_springbean.xml");**

**D- ApplicationContext applicationContext=new**

**ClassPathXmlApplicationContext("classpath:com/accenture/lkm/resource/ my\_springbean.xml"); (Right)**

**61) Which of the following is/are CORRECT with respect to Spring transaction methods?**

(i) If a method throws runtime/unchecked exception, then rollback will be triggered implicitly

(ii) If a method throws checked exception, then rollback will be triggered implicitly (if not rollback then this is right)

Choose the most appropriate option

**Ans:-B- Only i**

**62) Service class(Physical Transaction)**

**//=====**

**@Transactional(value="txManager", RollbackFor=InvalidDummyException.class)**

**public Integer addEmployeeAndDepartment(EmployeeBean employeeBean, DepartmentBean departmentBean) throws Exception{**

**int result=0;**

**int deptId=employeeDepartmentDAO.addDepartment(departmentBean);**

**employeeBean.setDepartmentCode(deptId);**

**result= deptId+employeeDepartmentDAO.addEmployee(employeeBean);**

**return result;**

**}**

**//DAO Class(Logical Transactions)**

**//=====**

**@Transactional(value="txManager", RollbackFor=InvalidDummyException.class)**

**public Int addEmployee(EmployeeBean employeeBean) throws Exception {**

**// code for creating employee**

**Throw new InvalidDummyException("There is an error");**

**}**

```
//Line -X
public Int addDepartment(DeaprtmentBean departmentBean) throws Exception {
// code for creating Department
}
```

Assume InvalidDummyException is checked exception.  
Choose from below the valid option to be placed at Line-X so that rollback of physical transaction can rollback all the logical transactions except for addDepartment();

Ans:-

```
@Transactional(value="txManager",propogation=Propagation.REQUIRES_NEW)
@Transactional(value="txManager",propogation=Propogation,REQUIRES_NEW)
```

63) Which object is created by web container for each Servlet during servlet initialization  
Ans:-B- ServletConfig

64) Rohit wants to externalize the validation error messages. So, he defined the messages in 'com/accenture/lkm/resoruces/messages\_en.properties files'.Help rohit to configure the MesageSource beam in the context file by choosing a valid option from beio

Ans:-

```
<bean id="messageSource" class="org.springframework.support
<property name="name"
value="classpath\com\accenture\lkm\resources\messages_en.properties"/>
</bean>
```

65) Rohit deployed the application in the Apache Tomcat Server. But he got anexception as "org.springframework.beans.factory. NoSuchBeanDefinitionException:No qualifying vean of type [com.accenture.lkm.dao.PruchaseDAO]". Choose from below valid options tosuccessfully execute his application  
Ans:- A) Configure in the root context confifuration C) Configure in the child context configuration

66) Rohit wants to display all the validation error messages in purchase.jsp.Help him to display those messages at the bottom of the page.  
Choose from below a valid option. Asssume the below taglib directives are addedin the jSP.

```
<%@taglib url="http://springframework.org/tags/form" prefix="form"%>
Ans:-
<form:form method="POST" modelAttribute="bean" action="store.html">
<table border="3">
<!--Assume form elements are mentioned appropriately -->
</table>
<spring:hasBindErrors name="bean">
```

67) Consider the code given below:

```
package com.accenture.lkm;
public class Employee { private list<String>hobbies;
```



```
//Getters and Setters are already coded
}
<beans>
<bean id ="employee" class="com.accenture.lkm.Employees">
<property name="hobbies">
<list>
<value>Playing Cricket </value>
<value>Watching Movies </value>
</list>
</property>
</bean>
</beans>
public class UIT Tester { public static void main(String
.....
```

**Ans:- Employee 2 hobbies [Playing CricketWatching Movies]**

**68)** Rohit wants to validate the quantity entered by the customer as mentioned in requirement 2. Choose from the below valid option.[Choose 2]

**Ans:- Add @Range(min=1, max=10)anotation in the Bean class The request handler method should include @Valid to@ModelAttribute parameter**

**69)** Which method is used to retrieve the recent access time of a request by user from session? choose from below valid option.

```
Getid()
Getsession()
Getcreationtime()
GetLastAccessedTime()
```

**Ans:- D- getLastAccessedTime()**

**70)**Rohit wants to navigate to success.jsp page on successful submission. Assume he wrote a handler method to return ModelAndView with logical view name as success. He needs to configure a View Resolver bean in child configuration file to resolve the logical view name.Choose from below a valid option

**Ans:- <bean class="org.springframework>**

**<property name="prefix">**

**<value>/WEB-INF/jspViews/</value>**

**</property>**

**<property name="suffix">**

**<value>jsp</value>**

**</property>**

**</bean>**

```
<bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">
<property name="prefix" value="/WEB-INF/jspViews/" />
<property name="suffix" value=".jsp" />
</bean>
```

**71)** Rohit wants to create a request handler method in controller that can handle customer request and send the response back to the response back to the customer. Help Rohit to achieve this with a valid option given below.

**Ans:-** Create a request handler method in the controller to map the request using `@RequestMapping` annotation and return the `ModelAndView` object.

**72)** Rohit wants to auto populate the items available for the selected SportsType to the respective drop-down box. Assume he defined the respective method in the controller which can return Map. Rohit is supposed to bind the key as label attribute, value as value attribute in the drop-down box. Choose from below a valid option.

**Ans:-** A

**73)** Rohit wants to create an auto populated drop-down box in purchase.jsp page. Help Rohit to implement a proper handler method in controller that can retrieve sportsType defined in DAO layer. Choose from below a valid option.

**Ans:-** Create a method in the controller to invoke DAO layer method which returns the Map of SportsType with `@ModelAttribute` annotation.

**74)** From the following tables match the JSP implicit objects given in table A with APIs in sevlets package given in table B.

Table A	Table B
1)out	i)ServletContext
2)application	ii) JSPWriter
3)exception	iii) Object
4)page	iv) Throwable

**Ans:-A-** 1-ii, 2-i, 3-iv, 4-iii

**75)** Which of the following statements(s) is/are CORRECT with respect to JSP?

(i)Request Scope variable is available in the page where it is declared as well as whenever the request object is shared.

(ii)classRequest Dispatcher of servlet API shares the request and response object to the forwarded or included page. Choose the most appropriate answer

**Ans:-both i and ii**

**76)** Assume that class "Employee is created with property "empId" and its setter method.Refer the spring configuration given below. 1 2 3 How many Employee objects will be created if the below code executes successfully in a main method?Assume that "context" referes to spring container Application Context public static void main(String[] args) { // code for Application Context creation goes here Employee emp1 = (Employee) context.getBean("emp");Employee emp2 = (Employee) context.getBean("emp"); Employee emp3= emp1;

**Ans:-** 1

**77)** Blessie is a web developer wants to create a wep page which always lands in portal.accenture.com,whenever user hits on welcome button in index.html given below

Index.html

```
-----  
<form action="MyServlet" method="POST">  
    <input type="submit" value="welcome">  
</form>
```

Now Blessie started writing the code, all of a sudden, she handed over the job of completing it to you due to unavoidable reasons . Inompletee cpde is attached below foryour references

MyServlet.java

```
-----  
@WebServlet(urlPatterns="/MyServlet")  
Public class Myservlet extends HttpServlet {  
Private static final long serialVersionUID = 1L;  
  
Public MyServlet()  
{  
super();  
}  
Protected void doGet(HttpServletRequest request HttpServletResponse response) throws servletexception,  
ioexception {  
PrintWriter out=response.getWriter();  
response.setContentType("Text/html");  
/** Code is missing **/  
}  
Protected void doPost(HttpServletRequest request HttpServletResponse response) throws servletexception,  
ioexception {  
doGet(request, response);  
}}  
}
```

You must choose from below a VALID option to complete the code above:

Ans:- - B) response.sendRedirect(["https://portal.accenture.com"](https://portal.accenture.com));

78) Refer the below incomplete code for a test class

```
@RunWith(SpringJUnit4ClassRunner.class)  
@ContextConfiguration( ..... MyConfig.class) (profiles ..... "myProfile")  
public class TestSpringCustomClass {  
@Autowiredprivate Employee employee:  
// test cases goes here }
```

Choose from below a valid combination to complete the above code in the orderas it appears.

Ans:- A- classes,@ActiveProfiles

79) From the following identify which is NOT a session management technique?

**Ans:- A-requestDispatcher**

**80)** Which of the following is/are CORRECT statement(s) with respect to spring test?  
@ContextConfiguration can point to Spring XML configuration and JAVA configuration.  
@ActiveProfiles is used to activate one or more profiles for the test cases Choose the most appropriate option

**Ans:- Both i and ii**

**81)** Which of the following statements is/are CORRECT with respect to JSP?  
(i)Default scope of a variable is page  
(ii)Session scope variable will be available in all JSP files if it is accessed in a single session  
(iii) JSTL codes are placed inside service() method of translated servlet Choose the most appropriate option

**Ans:- both i and ii**

**82)** Which of the following statements is/are CORRECT with respect to JSP?  
(i)Default scope of a variable is requested  
(ii)Application scope variable will be available in all JSP files irrespective of session in which it is accessed  
(iii)JSTL codes are not placed inside service() method of translated servlet

**Ans:- both ii and iii**

**83)** Which of the following statement(s) is/are CORRECT with respect to JSP?  
(i)Default scope of a variable is application  
(ii)JSTL codes are placed inside service() method of translated servlet  
(iii)Page scope variable is available only on the page where it is declared  
Choose the most appropriate option

**Ans:- C - Only iii**

**84)** Which of the following is correct with respect to propagation type  
Propagation.REQUIRES\_NEW?  
i) For a method, begin a new transaction always  
ii) If a method is invoked from another method and a transaction exists, then the same transaction is used.  
Choose the most appropriate option

**Ans:- only ii or only i**

**85)** Miss Rose has to create a HomeServlet to store the username and mobile number in session objects while processing registration request. The registration html page is the following

```
<html>
<head><title>Registration Page</title></head>
<body>
<form action="Register" method="post">
<label>Name</label>
```

```
.....  
<input type="submit" value="Register">  
</body></html>
```

Help her to complete the code in RegistrationServlet by identifying the right method.

**Ans:- A = Protected void doPost(HttpServletRequest ..... Throws ServletException,IOException {**  
**//TODO Auto-generated method...**  
**String name=request.getParameter("...");**  
**String mobile=request.getParameter(".....");**  
**HttpSession session=request.getSession();**  
**Session.setAttribute("username",name);**  
**Session.setAttribute("number",mobile); }**

**86)** Consider the following code

```
Login.html;  
<html><body>  
<h2>Login Page</h2>  
<form action="SessionServlet">  
<table>  
<tr><td>UserName</td><td><input type="text" name="uname"/></td></tr>  
<tr><td>Password</td><td><input type="password" name="pwd"/></td></tr>  
</table>  
<input type="submit" value="Login"/>  
</form></body></html>
```

Consider user is entering "John" in the username field and "admin" in the password field then he is submitting the form

```
SessionServlet2.java:  
Package.com.accenture.lkm;  
Import java.io.*;  
Import javax.servlet.*;  
Import javax.servlet.http.*;  
Public class SessionServlet2.....  
.....  
.....
```

What will be the outcome on clicking the link "Next Page" in SessionServlet?

**Ans:- Page 2 Welcome null .....** Without the code of the SessionServlet2 class it is impossible to determine what the outcome

**88)** Assume that class "Employee" is created with property "empld" and its setter me Refer the spring configuration given below

1 <bean id='emp' class con Employee'>

2 <property name "espid" value John

How many Employee objects will be created if the below code executes successfully method? Assume that "context" refers to spring container ApplicationContext

**Ans :-A- 1**

**89)** Refer the Spring configuration code given below:

@Configuration

public class AppConfig {

@Bean

public Address createAddress( ) {

Address address = new Address( ) :

return address ;

}}

Identify the correct code to create spring container object and get the address object injected. Choose the most appropriate option

**Ans :- B -** `ApplicationContext ctx = new AnnotationConfigApplicationContext(AppConfig.class);`

`Address address = ctx.getBean(Address.class);`

**ApplicationContext ctx =**

**new AnnotationConfigApplicationContext(AppConfig.class);**

**Address address=(Address)ctx.getBean("Address");**

**90)** Refer the incomplete code given below:

package com.accenture.lkm;

public class Job

{

private int jobId;

//getter and setter

}

public class Person

{

private String[] skills;

\_\_\_\_\_line1\_\_\_\_\_

private Job job;

person(String[] skills){

this.skills.=skills;

}

//getter and setter

}

Refer the Springconfig java class below

\_\_\_\_\_Line2\_\_\_\_\_

public class Springconfig{

\_\_\_\_\_Line3\_\_\_\_\_

```
Public person getperson(){  
String skills=("java","spring core");  
Return new person(skills);
```

Ans:-

```
@value("#jobObjects") @configuration, @Bean, @beanname ="my...");
```

OR

```
@Value("#(myjob)"1 @configuration @Bean @Beanvalue ="myjob")
```

OR

```
@value("#jobObjects") @configuration @bean
```

Line 1:

```
@Value("#{jobObjects}")
```

Line 2:

```
import org.springframework.context.annotation.Bean;  
import org.springframework.context.annotation.Configuration;  
import org.springframework.beans.factory.annotation.Value;
```

Line 3:

```
@Configuration  
@Bean(name = "myjob")  
public Job getJob(){  
return new Job();  
}
```